






**BAYESIAN NETWORK**

**Patent number:** JP2001075808  
**Publication date:** 2001-03-23  
**Inventor:** SKAANNING CLAUS; JENSEN FINN V; KJAERULFF UFFE; PELLETIER PAUL A; JENSEN LASSE ROSTRUP; PARKER MARILYN A; BOGORAD JANICE L  
**Applicant:** HEWLETT PACKARD CO  
**Classification:**  
- **international:** G06F9/44; B41J29/46; G06F3/12  
- **european:**  
**Application number:** JP20000206363 20000707  
**Priority number(s):**

AN

**Also published as:**

 EP1069487 (A1)  
 US6535865 (B1) AL  
 EP1069487 (B1)  
 DE60007587T (T2)  
 DE60007587D (T2)

**Report a data error here****Abstract of JP2001075808**

**PROBLEM TO BE SOLVED:** To obtain a knowledge for troubleshooting by repairing the cause which can be represented by one of coupled cause nodes in a troubleshooting step and representing respective steps by a plurality of 1st troubleshooting nodes.

**SOLUTION:** A problem to be troubleshot is discriminated (900) and the cause of the problem is discriminated (901). Then secondary cause is discriminated and the troubleshooting step (procedure) of the problem is discriminated (902, 903). Then the cause and troubleshooting step are made to correspond to each other and it is verified whether or not new cause or secondary cause is discriminated (904 to 906). Through this verification, the probability action of the cause and secondary cause and the probability of a question are evaluated (908) and the cost of the action and question is evaluated (909). Thus, an action and a question which require special control are discriminated and displayed with a plurality of troubleshooting nodes (910).

---

Data supplied from the esp@cenet database - Patent Abstracts of Japan

BEST AVAILABLE COPY

AN

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-075808

(43)Date of publication of application : 23.03.2001

(51)Int.Cl.

G06F 9/44

B41J 29/46

G06F 3/12

(21)Application number : 2000-206363

(71)Applicant : HEWLETT PACKARD CO &lt;HP&gt;

(22)Date of filing : 07.07.2000

(72)Inventor : SKAANNING CLAUS  
JENSEN FINN V  
KJAERULFF UFFE  
PELLETIER PAUL A  
JENSEN LASSE ROSTRUP  
PARKER MARILYN A  
BOGORAD JANICE L

(30)Priority

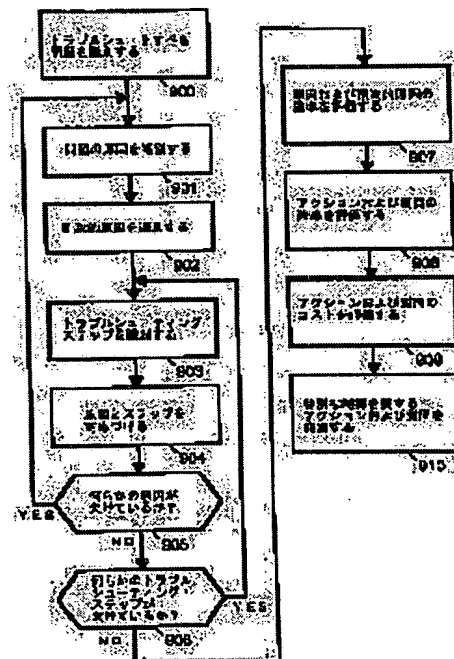
Priority number : 99 353727 Priority date : 14.07.1999 Priority country : US

## (54) BAYESIAN NETWORK

(57)Abstract:

PROBLEM TO BE SOLVED: To obtain a knowledge for troubleshooting by repairing the cause which can be represented by one of coupled cause nodes in a troubleshooting step and representing respective steps by a plurality of 1st troubleshooting nodes.

SOLUTION: A problem to be troubleshot is discriminated (900) and the cause of the problem is discriminated (901). Then secondary cause is discriminated and the troubleshooting step (procedure) of the problem is discriminated (902, 903). Then the cause and troubleshooting step are made to correspond to each other and it is verified whether or not new cause or secondary cause is discriminated (904 to 906). Through this verification, the probability action of the cause and secondary cause and the probability of a question are evaluated (908) and the cost of the action and question is evaluated (909). Thus, an action and a question which require special control are discriminated and displayed with a plurality of troubleshooting nodes (910).



## LEGAL STATUS

[Date of request for examination]

25.06.2003

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

AN

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2001-75808

(P2001-75808A)

(43)公開日 平成13年3月23日(2001.3.23)

(51)Int.Cl.<sup>7</sup>

識別記号

F I

テーマコード\*(参考)

G 0 6 F 9/44

5 5 0

G 0 6 F 9/44

5 5 0 C

B 4 1 J 29/46

B 4 1 J 29/46

A

G 0 6 F 3/12

G 0 6 F 3/12

D

K

審査請求 未請求 請求項の数 1 O L (全 46 頁)

(21)出願番号 特願2000-206363(P2000-206363)

(22)出願日 平成12年7月7日(2000.7.7)

(31)優先権主張番号 0 9 / 3 5 3 7 2 7

(32)優先日 平成11年7月14日(1999.7.14)

(33)優先権主張国 米国 (U S)

(71)出願人 398038580

ヒューレット・パカード・カンパニー

HEWLETT-PACKARD COM  
PANY

アメリカ合衆国カリフォルニア州パロアル

ト ハノーバー・ストリート 3000

(72)発明者 クラウス・スカーニング

デンマーク、ディーケー9330、ドロニン

ランド、ホルテガードスエフ 1

(74)代理人 100081721

弁理士 岡田 次生

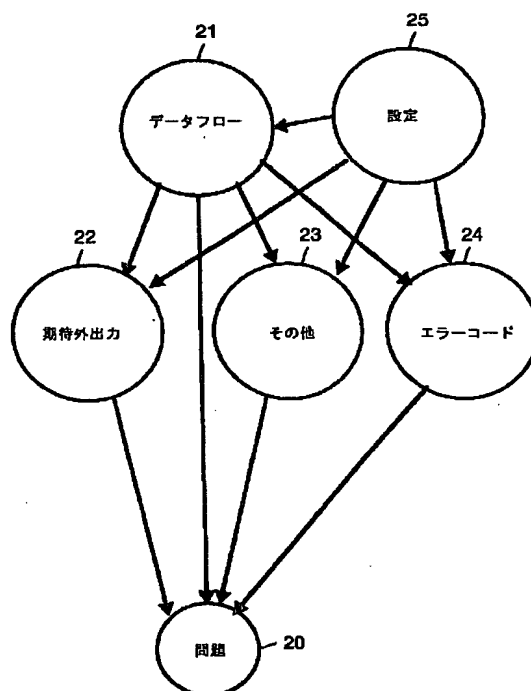
最終頁に続く

(54)【発明の名称】 ペイジアン・ネットワーク

(57)【要約】

【課題】 システムのトラブルシューティングを行う自動  
トラブルシュート機構を提供する。

【解決手段】 自動トラブルシュート機構はシステムのト  
ラブルシューティングを行うためペイジアン・ネットワ  
ークを使用する。システムのトラブルシューティングを  
行う準備として知識獲得が実行される。トラブルシュ  
ートすべき問題、その問題の原因および原因の副次的原因  
が識別される。原因および副次的原因に対応づけられた  
トラブルシューティング・ステップが識別される。識別  
された原因および副次的原因について確率が評価され  
る。アクションおよび質問のセットについて確率が評価  
される。アクションおよび質問についてコストが評価さ  
れる。この結果、トラブルシューティングに際して、問  
題を解決できる確率が最も高くコストの最も低いアク  
ションから順次、問題が解決されるまで、ユーザに適切  
なアクションが提案される。



## 【特許請求の範囲】

【請求項1】 システムの故障を引き起こすシステム・コンポーネントをモデル化するベイジアン・ネットワークであって、  
前記システム・コンポーネントが故障を引き起こしているか否かを示す状態を有する標識ノードと、  
前記標識ノードに結合され、故障を生成するシステム・コンポーネントの原因をそれぞれが表す複数の原因ノードと、  
前記複数の原因ノードのうちの少なくとも1つの原因ノードにそれぞれが結合され、該結合された原因ノードのいずれかによって表される原因を修復するアクションを提案するトラブルシューティング・ステップをそれぞれが表す複数の第1のトラブルシューティング・ノードと、を備えるベイジアン・ネットワーク。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、プリンタのサポート、特にベイジアン・ネットワークを使用するプリンタ・システムの自動診断に関する。

## 【0002】

【従来の技術】 現在、プリンタ製造業者がその顧客のシステムを診断するのにかかる費用は、非常に高価である。典型的には、顧客は製造業者のプリンタ電話サポート担当者呼び出す。この担当者は、問題の解決または原因の特定へ導くトラブルシューティングのシーケンス(すなわち一連の手順)を利用して顧客を誘導する。この方法は高いコストを伴う担当者の介入を必要とする。サポート担当者を使用する場合、プリンタ製造業者は多数のサポート担当者を雇い、顧客はそのプリンタ・システムに問題が発生すると担当者に電話する。担当者は、電話による顧客へのインタビューによって可能な限り多くの情報を収集しようと努める。担当者は、結論に到達する時、問題を解決または原因を特定したか、あるいは顧客先で問題を解決するため現場サポート担当者を顧客に派遣することとなる。

【0003】 電話サポート担当者の使用の1つの欠点は費用である。それに加えて、異なる担当者によって使用されるトラブルシューティング手順の順序および種類における整合性について問題が存在する場合がある。同じ問題について同じ順序で同じトラブルシューティング手順が顧客に提供されなければ、顧客は困惑するかもしれない。そのような整合性の欠如は問題である。また、電話サポート担当者の情報記録は限られており、データ収集プログラムを統合するには限界があり、またマルチメディア表現を合するにも限界がある。しかしながら、コンピュータシステムが容易に検出することができないような目立たない情報、例えば、顧客が質問にいらしているかどうか、顧客の経験のレベル等々のような情報を電話サポート担当者であれば検出することができること

は明らかであるので、電話サポート担当者の使用は、担当者と顧客の間に人と人のコミュニケーションが存在するという利点を提供する。

【0004】 デシジョン・ツリーを使用して、プリンタ・システムの自動診断を行うことができる。デシジョン・ツリー手法は、可能なトラブルシューティング手順をいわゆるデシジョン・ツリーとして規定する。ツリーの分岐のそれぞれにおいて、最後のステップで顧客によって提供された情報に基づいていくつかの分岐のうちの1つが選択される。しかしながら、実用上の理由からトラブルシューティング手順のうちの限られた数の可能なシーケンスだけが利用できるにすぎないという意味で、デシジョン・ツリーは静的である。デシジョン・ツリーの場合、顧客が利用することができなければならないすべてのシーケンスが符号化されなければならない。デシジョン・ツリーのサイズはそのようなシーケンスの数の指数関数であるので、限られた数のシーケンスを符号化することができないにすぎない。これは、平均するとデシジョン・ツリーのトラブルシューティングのシーケンスを長くすることになり、それにもかかわらず、すべての可能なシナリオを考慮に入れることができないため、問題を実際に診断する確率を低下させる。

【0005】 また、ケース型推論(すなわちケースに基づく推論)を使用して、プリンタ・システムの自動診断を行うことができる。ケース型手法は、種々の問題が観察されるトラブルシューティング・シナリオから大量の記述的ケースを収集する。現状に関する情報に基づいて、ケース型推論エンジンが、最も類似しているケースを選択することができる。最も類似するケースを調べて、可能な限り多くのケースを除外できる可能性が最も高い最良の次のアクションまたは質問が見出される。現状に最も一致する1つのケースが求められるまで、この手順が続けられる。

【0006】 ケース型システムは、トラブルシューティング領域を記述するケースを収集し、それらのケースを使用して、できるだけ迅速に範囲を1つのケースに狭めるアクションおよび質問を提案する。ケース型システムの品質は、そのケース・データベースに依存する。このデータベースは、プリンタ・システム領域を適切に記述することができるように非常に大規模でなければならない。すべての変数が2進数であるとすれば、プリンタ・システムにおける可能な構成/ケースは少なくとも2<sup>1,000</sup>ある。ケース型システムにとって役立つように十分記述的であるようにするには、これらケースから取り出されるケースのサブセットが極めて大きくなければならない。このように、多くの変数を有する印刷システムを最適レベルの精度で表す上で、ケース型システムが成功することは疑わしい。

【0007】 更に、ルール型システム(すなわち規則に基づくシステム)を使用して、プリンタ・システムの自

動診断を行うことができる。ルール型システムは、ベイジアン・ネットワーク (Bayesian network) で表現されることができるので、ベイジアン・ネットワークのサブセットとして知覚されることができる。ルール型システムは、ベイジアン・ネットワークのモデル化能力のサブセットを持ち、その信念 (belief) 更新方法は、それがベイジアン・ネットワークに関するものであるようには正確さが保証されない。

【0008】しかしながら、ルール型システムは、モデルにループがある場合には適していない更新方法を持つ。ループは、現実のシステムのモデル (例えば共通の原因、いくつかの原因を修復する手順など) では非常に一般的である。

【0009】ベイジアン・ネットワークに基づくトラブルシューティング機構の1つが、"Communications of the ACM, 38:49-57(1995)" のD. Heckerman, J. Brees およびK. Rommelse著の"Decision-theoretic Trouble shooting" に記載されている (以下の記述において、この文献をHeckerman文献と呼ぶ)。

【0010】ベイジアン・ネットワークは、変数間の因果関係を表す有向非巡回グラフであり、親が与えられると、条件つき確率分布を変数に関連づけるものである。ベイジアン・ネットワークにおける確率を正確に更新するための効率的な方法が開発されている。その例は、S. L. LauritzenおよびD. J. Spiegelhalter著の"Local Computations with Probabilities on Graphical Structures and their Applications to Expert Systems, Journal of the Royal Statistical Society Series B, 50 (2):157-224(1988)"、および、F. V. Jensen, S. L. LauritzenおよびK. G. Olesen著"Bayesian Updating in Causal Probabilistic Networks by Local Computations, Computational Statistics Quarterly, 4:269-282(1990)"に記載されている。

【0011】ベイジアン・ネットワークにおける確率を正確に更新するための効率的な方法は、HUGINエキスパート・システムにおいて実施されている。HUGINエキスパート・システムは、S. K. Andersen, K. G. Olesen, F. V. JensenおよびF. Jensen著の"HUGIN - a Shell for Building Bayesian Belief Universes for Expert Systems, Proceedings of the Eleventh International Joint Conference on Artificial Intelligence(1989)"に記載されている。

【0012】ベイジアン・ネットワークは、確率理論を使用して問題領域をモデル化する方法を提供する。他に、に関する情報が与えられたとすると、問題のベイジアン・ネットワーク表現を使用して、変数のサブセットに関する情報を提供することができる。ベイジアン・ネットワークは、変数(ノード)のセットおよび有向エッジ(変数間の接続)のセットから構成される。各変数は、相互に排他的な状態のセットを持つ。変数は、有向エッジ (di

rected edge) と共に、有向非巡回グラフ (directed acyclic graph) の頭文字を取ってDAGと呼称される) を形成する。親 $w_1, \dots, w_n$ を持つ各変数 $v$ について、条件つき確率テーブル $P(v | w_1, \dots, w_n)$ が定義される。 $v$ が親を持たない場合に、このテーブルが、周辺確率 $P(v)$ に減少することは明らかである。

【0013】ベイジアン・ネットワークは、医療診断、系統解析、計画、負債検出、ボトルネック検出などのような不確実性を有する多くのアプリケーション領域において使用されてきたが、その主要な領域の1つは診断である。診断(すなわち徴候を再発させる病気/誤動作の原因となる根元的要因)は、ベイジアン・ネットワークのモデル化技術に非常に向いている。

【0014】ベイジアン・ネットワークの正確な信念更新のための現在最も効果的な方法は、ネットワークをいわゆる接合ツリー (junction-tree) に変換する接合ツリー方法である。この方法は、F. V. Jensen, S. L. LauritzenおよびK. G. Olesen著"Bayesian Updating in Causal Probabilistic Networks by Local Computations, Computational Statistics Quarterly, 4:269-282(1990)"に記載されている。接合ツリーは、基本的には、1つのツリーが獲得され(すなわちすべてのループが取り除かれる)、クラスタの大きさが可能な限り小さくなるように変数をクラスタリングする。その後、このツリーにおいて、観察された変数が与えられたとすると、メッセージ受渡し方式は、すべての観察されなかった変数の信念を更新することができる。ベイジアン・ネットワークの正確な更新はNP-hard (NP困難) であるが(参照: G. F. Cooper著"The Computational Complexity of Probabilistic Inference using Bayesian Belief Networks, Artificial Intelligence, 42:393-405, (1990)"、ベイジアン・ネットワークのクラスのうちいくつかに対してはなお効果的である。印刷システムのためのネットワークは、数千という変数と多くのループを含んでいるが、合理的に効率的な信念更新を備える接合ツリーになお変換することができる。

【0015】上記引用Heckerman文献は、ベイジアン・ネットワークに基づく順次トラブルシューティングを実行する方法を提示する。

【0016】変数 $c_1, \dots, c_n$ によって表される $n$ 個のコンポーネントを持つトラブルシュートに対する装置について、Heckerman文献は、厳密に1つのコンポーネントが故障し、このコンポーネントが問題の原因であることを必要とする単一故障仮定に従う。情報の現在状態が与えられたとして、コンポーネント $c_i$ が異常である可能性を $p_i$ で表すとすれば、単一故障仮定の下では次の式が成り立つ。各コンポーネント $c_i$ は、(時間および(または)金額で計測される)観察コスト $C_i^o$ および修復コスト $C_i^r$ を持つ。

【0017】

【数1】

$$\sum_{i=1}^n p_i = 1$$

【0018】ここでは記述しないが、いくつかの付加的な緩和された前提の下では (Heckerman文献参照)、現在の情報で更新される故障確率  $p_i$  に関して、最も高い比率  $p_i/C_i$  を有するコンポーネントを観察することが常に最適であることがわかる。このことは、比率が、故障 \*

\*の確率と観察コストのバランスをとり、最も高い故障確率と最も低い観察コストを持つコンポーネントを示すので、直感的に理解できる。このように、単一故障仮定の下では、最適な観察-修復シーケンスが、次の表1に提示される手順によって与えられる。

【0019】

【表1】

ステップ1	装置が機能していなければ、コンポーネントの故障確率 $p_i$ を計算する
ステップ2	最も高い比率 $p_i/C_i$ を持つコンポーネントを観察する
ステップ3	コンポーネントが故障していれば、修理する
ステップ4	コンポーネントが修理されたならば終了し、そうでなければステップ1へ進む

【0020】上記表1の手順において、ステップ3でコンポーネントが修理される場合には、その装置が修理されなければならないことは単一故障仮定から自ずとわかることであり、よってトラブルシューティング・プロセスを停止することができる。単一故障仮定が採用されれば、このアルゴリズムは十分合理的に作用し、この場 \*

※合、ステップ1がステップ2およびステップ3で得られた新しい情報を考慮に入れ、ステップ4が次の表2に示されるように置き換えられる。

【0021】

【表2】

ステップ1	装置が機能していなければ、コンポーネントの故障確率 $p_i$ を計算する
ステップ2	最も高い比率 $p_i/C_i$ を持つコンポーネントを観察する
ステップ3	コンポーネントが故障していれば、修理する
ステップ4	装置がなお誤作動すれば、ステップ1へ進む

【0022】Heckerman文献は、最適なトラブルシューティング・シーケンスの期待されるコストが (援助を求めるための製造業者呼び出しのような) サービス呼び出しのコストより高いときに使用されるサービス呼び出しを取り扱う理論を紹介している。この理論は、複数の故障および非基本的観察を持つシステムを近似的に取り扱うことを可能にするように、上記手順を変更する。非基本的観察とは、コンポーネントではないがトラブルシューティング処理のため有益な情報を潜在的に提供するものについての観察のことである。J. S. Breese, D. Heckerman 著 "Decision-theoretic Troubleshooting: A Framework for Repair and Experiment, Technical Report MSR-TR-96-06, 1996) Microsoft Research, Advanced Technology Division, Microsoft Corporation, Redmond, USA" には、最適トラブルシューティング・シーケンスのコストを潜在的に下げることができ一層有益な情報を提供するようにシステムの構成変更を可能にする一層改善された方法が記述されている。

【0023】

【発明が解決しようとする課題】しかしながら、上記 Heckerman 文献に記載されているベイジアン・ネットワークに基づくトラブルシューティングの方法は、原因とアクションの間が1対1の対応関係にあるという、実際に

は保たれない関係を有しており、また質問の近視的 (1ステップ先読みの) 選択を行うので、質問が多数存在する時には質問の選択が極めて遅くなる。更に、上記 Heckerman 文献は、そのトラブルシューティングのための知識獲得の方法を提示していない。

【0024】

【課題を解決するための手段】発明の課題を解決するため、システムの故障を引き起こすシステム・コンポーネントをモデル化するベイジアン・ネットワークが提供される。該ベイジアン・ネットワークは、当該システム・コンポーネントが故障を引き起こしているか否かを示す状態を持つ標識ノードと、上記標識ノードに結合され、故障を生成するシステム・コンポーネントの原因をそれぞれが表す複数の原因ノードと、上記複数の原因ノードのうちの少なくとも1つの原因ノードにそれぞれが結合され、該結合された原因ノードのいずれかによって表される原因を修復するアクションを提案するトラブルシューティング・ステップをそれぞれが表す第1の複数のトラブルシューティング・ノードとを備える。

【0025】本発明の好ましい実施形態によると、システムのトラブルシュートを実行する準備段階において知識獲得が実行される。トラブルシュートすべき問題が識別される。該問題の原因 (複数) が識別される。該原因の

副次的原因(複数)が識別される。トラブルシューティング・ステップが識別される。トラブルシューティング・ステップは、原因および副次的原因に対応づけられる。識別された原因および副次的原因について確率が評価される。アクションおよび質問のセットについて確率が評価される。アクションおよび質問のセットについてコストが評価される。

【0026】また、本発明の好ましい実施形態によると、特別な取り扱いを必要とするアクションおよび質問が識別される。問題の原因を識別するのに領域専門家が使用される。トラブルシューティング・ステップは、原因または副次的原因のいずれかを解決することができるアクションと、原因または副次的原因に関する付加的情報を提供する質問とを含む。アクションを含むトラブルシューティング・ステップのそれぞれは、アクションが解決することができる原因または副次的原因のいずれかに対応づけられ、質問を含むトラブルシューティング・ステップのそれぞれは、質問が関連する原因または副次的原因のいずれかに対応づけられる。

【0027】アクションおよび質問についてコストを評価する時、関連する原因または副次的原因のそれぞれについて、第1のアクションの実行が問題を正しく解決するか否かに関する判断が行われる。それに加えて、顧客が第1のアクションを正しく実行する可能性に関する判断が行われる。第1のアクションについてのコストは、第1のアクションを実行する時間、第1のアクションを実行する場合に何かを破壊するリスク、第1のアクションを実行するために必要な部品を購入するために必要とされる金額、および第1のアクションが提案される時にユーザが侮辱と感じるかもしれない度合いを考慮する因子を含むことができる。

【0028】本発明に従うトラブルシュート機構は、ベイジアン・ネットワークを利用する。例えば、システムの故障を引き起こすシステム・コンポーネントをモデル化するベイジアン・ネットワークは、標識ノード、複数の原因ノード、および第1の複数のトラブルシューティング・ノードを含む。標識ノードは、そのシステム・コンポーネントが故障を引き起こしているか否かを標示する状態を有する。各原因ノードは、故障を生成するシステム・コンポーネントの原因を表す。各トラブルシューティング・ノードは、トラブルシューティング・ステップを表す。各トラブルシューティング・ステップは、該トラブルシューティング・ノードが結合されている原因ノードのいずれかによって表される原因を修復するアクションを提案する。原因ノードは、システム・コンポーネントの故障の原因に対する確率分布を表す。

【0029】ベイジアン・ネットワークは、更に、質問ノードを含むことができる。質問ノードは、質問が回答される時、当該質問ノードが結合されている原因ノードによって表される原因に関する潜在的情報を提供する質

問を表す。また、質問ノードは、システム・コンポーネントの徴候または故障原因に必ずしも関係しない一般的なタイプの質問を表すこともできる。

【0030】第1のアクションを提案する第1のトラブルシューティング・ステップに関して、第1のアクションが第1の原因を解決するか否かを計算する時、不確実性因子を利用することができる。不確実性因子は、ユーザが間違っ第1のアクションを実行する確率を表す。好ましい実施形態において、複数の原因ノードのうちの少なくとも2つの原因ノードは、共通の副次的原因を持つことができる。

【0031】例えば、トラブルシューティング機構は、5つのベイジアン・ネットワークを利用する印刷診断システムである場合、第1のベイジアン・ネットワークは、印刷しようとする時に顧客がプリンタからの出力を得られない場合のすべてのエラー、および顧客が損なわれた出力を受け取る場合のすべてのエラーを取り扱う。第2のベイジアン・ネットワークは、顧客が、期待していない出力を得るすべてのエラーを取り扱う。第3のベイジアン・ネットワークは、プリンタのコントロール・パネル上で見ることができるエラーコードのすべてのタイプを取り扱う。第4のベイジアン・ネットワークは、第1のベイジアン・ネットワーク、第2のベイジアン・ネットワークおよび第3のベイジアン・ネットワークによってカバーされないプリンタのその他の誤った動作を取り扱う。第5のベイジアン・ネットワークは、印刷システムにおけるプリンタに関するすべての可能な設定を表す。

【0032】本発明は、知識獲得の効率をできる限り低く維持し、診断プロセスの品質をできる限り高くするよう、両者を組み合わせる新しいいくつかの概念を提示する。

【0033】コストの正確な評価と組み合わせて使用されるトラブルシューティングのアクションおよび質問を選択するための手段は、この発明の好ましい実施形態が、可能な限り少ないステップで診断に到達することを可能にする。

【0034】本発明に従う自動トラブルシューティング機構によって、顧客システムからプログラムによって取得される情報およびトラブルシューティング・セッション(診断の成功または失敗)の結果を含めて、顧客のトラブルシューティング・シーケンスの記録を容易にすることができる。この情報のすべては、人手を介することなく自動トラブルシューティング機構によって直接記録されることができる。

【0035】さらに、自動トラブルシューティング機構は、データ収集プログラムを容易に組み入れることができる。診断の速度を上げるためにトラブルシューティング・プロセスにおいて使用することのできる適切なデータのために、顧客のPC、プリンタ等を直接検索するデ

ータ収集プログラムを使用して、顧客からのデータの対話的収集を改良することは比較的容易である。

【0036】また、本発明の好ましい実施形態に従った自動トラブルシューティング機構は、マルチメディア表現の組み込みを容易にする。画像やサウンドのようなマルチメディア表現を利用して、顧客に問題を例示して顧客を正しい選択に導くことが可能である。多くの状況において、画像は、特に問題シナリオの記述を単純化し、顧客がより一層正しい選択を行うようにすることができる。

【0037】本発明の好ましい実施形態は、非常に効率的なやり方で自動トラブルシューティング機構を構築し、次に、明確に定義されたプロセスを実行する知識獲得(すなわちオーサリング)方法を提示する。知識獲得は、通常面倒で非常に時間を浪費するものであるため、一般に自動トラブルシューティング機構のボトルネックと見なされている。本発明の好ましい実施形態に従った自動トラブルシューティング機構は、一般的なベイジアン・ネットワークのモデル化フェーズに制約を課し、非常に厳格で単純な構造だけを許容することによって、知識獲得の範囲を制限し、その効率を増大させる。

【0038】本発明の好ましい実施形態に従った自動トラブルシューティング機構は、その他いくつかの利点を持つ。トラブルシューティングがコンピュータ・プログラムによって制御されるので、トラブルシューティング機構とユーザの間で発生するすべてを記録することが可能である。トラブルシューティング機構が問題を解決することができない状況においては、トラブルシューティング・プロセスを引き継ぐことができる経験豊かなサポート担当者に制御を与えることができる。この担当者は、以前に提案され実行されたステップ、および原因に関する最終的な確率の記録を見ることができる。サポート担当者は、この情報を使用して、スキップされたステップを再度提案すべきか否か、疑わしい回答を持つ実行済みステップを再度提案すべきか否か、またはトラブルシューティング機構に含められていない一層進んだステップを提案すべきか否かを決定することができる。このようにして、自動トラブルシューティング機構は、サポート担当者呼び出す回数を少なくするだけでなく、サポート担当者が取り扱うことができないいくつかのケースにおいてサポート担当者を援助する。

【0039】すべての情報の記録によって、いわゆる学習技法を使用して、トラブルシューティング・モデルの確率およびコストの一層の調整が可能となる。ベイジアン・ネットワークの分野に精通している者にとって容易に明らかであるように、質問、アクションおよび原因の確率は、大量に収集される情報、たとえば識別された原因、質問に対する回答、成功または失敗したアクション等によって改善されることができる。また、提案から回答までの時間間隔を単に測定することによって、アクシ

ョンおよび質問のコストの時間的要素を改善することも可能である。これは、多数のケースについて平均し、ステップの真の時間的要件を生成する。

【0040】本発明に従ってベイジアン・ネットワークと共に構築される自動化トラブルシューティング機構は、プリンタ・システムの高価な診断に関連する問題を解決する。本発明によって、ユーザは、多数の問題そのものを診断し、サポート担当者への電話を節約し、費用を減らすことができる。

10 【0041】また、本発明は、診断の品質および速度を改善する。本発明は、最適の(所定の条件のもとでは、できるだけ短く)トラブルシューティング・シーケンスを生成する。

【0042】また、本発明は、診断の整合性を改善する。本発明は、トラブルシューティング機構の構築者のモデル化柔軟性を制限し、ベイジアン・ネットワークに関するどのような知識を持つことなく実行される、明確かつ最良に構築された知識獲得プロセスを定義することにより、多くの診断プロジェクトにおいて観察される知識獲得ボトルネックを取り除くことを可能にする。

20 【0043】上記引用Heckerman文献によって提案された自動トラブルシューティング機構と比較して、本発明は、いくつかの局面を改善し、拡張する。本発明は、従来技術に見られない、トラブルシューティング機構モデルの知識獲得の完全な方法を含む。また、本発明は、いくつかの領域における最善な次のステップを選択するアルゴリズムをもカバーする。

【0044】

30 【発明の実施の形態】図1は、本発明の1つの好ましい実施形態に従うトラブルシューティング環境の概要を示している。図1は、ウェブ・サーバ200、顧客パーソナル・コンピュータ(PC)205、プリンタ・サーバ209およびプリンタ210を示す。プリンタ・システムのトラブルシューティング機構201は、ウェブ・サーバ200上で稼働する。顧客PC205のユーザは、インターネット202を経由してトラブルシューティング機構201にアクセスすることができる。ウェブ・サーバ200にアクセスするため、顧客PC205の範囲内のウェブ・ブラウザ206が使用される。トラブルシューティング機構201との顧客の対話に応答して、トラブルシューティング機構201は、顧客が実行することができるトラブルシューティング・ステップについての提案203で回答する。トラブルシューティング機構201は、本質的には、人工知能を利用するエキスパート・システムとして機能する。顧客は、提案203に対して行動したことによる結果をトラブルシューティング機構201に知らせる情報204を、トラブルシューティング機構201に送り戻す。情報204は、顧客がプリンタ・サーバ209から得る情報207、および(または)顧客がプリンタ210から得る情報208を含むことができる。

50 【0045】図2は、ウェブ・サーバ200の概略的なブ



ロック図を示す。トラブルシューティング機構201は、ウェブ・サーバ200のメモリ301において動作する。トラブルシューティング機構201は、トラブルシューティング・モデルを記憶するため二次記憶装置303を利用する。トラブルシューティング処理を監視し、トラブルシューティング・モデルを保守するために、表示装置304を専門家によって使用することができる。ウェブ・サーバ200は、また、キーボードのような入力装置305、CPU306、および、顧客PC205のウェブ・ブラウザ206と通信するためのネットワーク・カード307を備える。

【0046】図3は、トラブルシューティング処理のコンポーネントの概要である。ウェブ・サーバ200が示されている。顧客は、顧客PC401で稼働しているウェブ・ブラウザ206を介してウェブ・サーバ200の範囲内の(図1に示されている)トラブルシューティング機構201と通信する。顧客は、トラブルシューティング機構201から提案203を受け取り、それに応答して回答204を送る。プリンタ・サーバ209およびプリンタ210から構成されるプリンタ・システムにおいて故障が発生した時、顧客はトラブルシューティング機構201を使用する。一般的には、顧客がアプリケーション406から印刷を行おうとする時、印刷ジョブは、まずプリンタ・ドライバ407に行き、次にローカル・スプーラ408を通過し(スプーラが使用される場合)、そしてオペレーティング・システム(OS)のリダイレクト(転送機能)409に進む。オペレーティング・システム(OS)リダイレクト409はオペレーティング・システムの一部であって、印刷ジョブが、ネットワーク・ドライバ410およびネットワーク・カード411を経由してネットワーク接続413に進むのか、または、ローカルにパラレル接続されたプリンタの場合におけるローカル・ポート412へ進むのか決定する。印刷ジョブがローカルにパラレル接続されたプリンタに進むとすれば、印刷ジョブは、プリンタ210に到達する前にパラレル・ケーブル415を通過する。印刷ジョブが、ネットワーク・プリンタへ進むと場合、ネットワーク接続413を通過してプリンタ・サーバ209へ達するか、または、直接ネットワーク接続414を経由してプリンタ210へ達する。直接ネットワーク接続414は、例えばヒューレット・パッカード社のHP LaserJet 5Siのような所定のプリンタに利用することができる。プリンタ210がプリンタ・サーバ209によって制御される時、印刷ジョブは、プリンタ・サーバ209のプリンタ待ち行列420を通過し、その後、プリンタ210がプリンタ・サーバ209にどのようにして接続されているかによって、ネットワーク接続417またはパラレル・ケーブル418のいずれかを經由してプリンタ210に達する。

【0047】アプリケーション406、プリンタ・ドライバ407、スプーラ408およびOSリダイレクト409のすべては、顧客PC205上のオペレーティング・システム405において動作する。アプリケーション406から印刷ジョブ

を印刷する時、印刷ジョブは、システム設定に従って、上記経路の1つをたどってプリンタ210へ達する。その経路上で何かしらの誤りが起きれば、その結果、出力がないか、または期待外の出力となる。トラブルシューティング機構201は、システムにおけるコンポーネントの検査を通して、どのコンポーネントが問題を引き起こしたかを判断しようとする。

【0048】図4は、トラブルシューティング機構201を実現するために知識獲得を実行するステップの概要を示している。知識獲得プロセスは、いわゆる領域専門家からその領域に関する十分な情報を収集することによってトラブルシューティング・モデルを構築するプロセスである。領域専門家は、モデル化されつつある領域に(このケースではプリンタ・システムに)精通している。このような領域専門家は、当該領域についての詳細な知識を持ち、製品の製造段階、トラブルシューティングまたはサポート段階において支援を行う。知識獲得プロセスは、そのプロセスのルールおよび必要条件に精通している誰かによって誘導されなければならない。ベイジアン・ネットワークにおいては、知識獲得プロセスへ参加または知識獲得プロセスを誘導するのに専門的知識を必要としない。説明をわかりやすくするため、図4に記載されるステップの説明を介した1つの例として、"薄い印刷(light print)"という問題を使用する。"薄い印刷"は、ユーザが、期待したものより濃さが薄い出力をプリンタから受け取るという問題である。

【0049】ステップ900において、トラブルシュートすべき問題が識別される。モデル化されつつある問題が識別され、精密に定義され、他の問題と区別される。最初に、検討中の問題およびトラブルシューティング・ツールの使用者を初期的に正確に定義することは非常に重要である。なぜならば、それが、引き続いて行われる知識獲得手順に大きな影響を与えるからである。原因および手順(ステップ)の両方を規定する時、使用者の技術レベルは重要である。なぜならば、エンドユーザによっては取り扱うことができないが、経験豊かなトラブルシューティング機構によっては取り扱うことができる原因および手順が存在するからである。以下の記述においては、プリンタ・システムの基礎的理解は持っているが、複雑な手順を実行するには誘導される必要があるエンドユーザを仮定する。

【0050】ステップ901において、問題の原因が識別される。このステップでは、領域専門家が検討対象の問題の原因を識別する。原因は、基本的には、その問題を引き起こす可能性のあるすべての異なるコンポーネント、プロパティ(property)またはイベントである。たとえば印刷問題を引き起こす仕様外の重力のようにあまりに稀で考慮する価値がない原因、または、たとえばプリンタ・コンポーネントに関する先進的な技術上の問題のようにユーザが関与することができない原因が存在す

10

20

30

40

50

るので、すべての原因を識別および規定することは通常不可能であるし、また必要でもない。そのような原因は、“その他の問題”と呼ばれる1つの漏洩(leak;リーク)原因にくくられるが、この漏洩原因は、更に、プリンタの電源の入り切りによって解決することができる“一時的問題”、およびユーザが解決することができない“永久的問題”という2つの副次的原因を含む。

【0051】原因を識別する際の困難な点の1つは、複数の原因を1つの原因に分類すべきか否か、または原因\*

\*を別々のものとすべきか否かの決定である。原因が別々に保持され、原因について異なるアクションが存在するならば、経験則からしてそれらアクションに関する知識獲得を行うことは比較的容易である。

【0052】例えば、“薄い印刷”という問題に関して、次の表3に示されているような原因および副次的原因が識別される。

【0053】

【表3】

原因/副次的原因	説明
用紙媒体	トナーが正しく付着しないようなタイプの用紙である場合、薄い印刷を引き起こす可能性がある
用紙送り経路の汚れ	用紙送り経路が汚れていれば、薄い印刷を引き起こす可能性がある。湿度、温度などの環境状況が極端な場合、それらすべてが薄い印刷の原因となり得る
トナー・カートリッジの問題	トナーカートリッジに関する問題、たとえばカートリッジのトナーが足りない場合、薄い印刷を引き起こす可能性がある
転写ローラの問題	転写ローラは、ドラム表面上のトナー・イメージが媒体上に転写されることを可能にするが、薄い印刷の原因ともなり得る。アプリケーションの不正確な設定—アプリケーション、プリンタ・ドライバおよびプリンタのコントロール・パネル上の設定が正しくなければ、明らかに、薄い印刷を引き起こす可能性がある。
プリンタ・ドライバの不正確な設定	
コントロール・パネルの不正確な設定	
損なわれたデータフロー	アプリケーションからネットワークを経由したプリンタへの流れのどこかで印刷ジョブが損なわれてわずかな変化があれば、印刷結果が期待していたものより薄くなる可能性がある
間違ったドライバの使用	プリンタに関する不正確なドライバの使用が薄い印刷を引き起こすことがある
その他の問題	上述のように、考慮する価値のない薄い印刷の原因があり、それらはこの項目に集められる

【0054】経験に従えば、このレベルで原因をモデル化することは、経験豊かな印刷システム電話サポート担当者によって使用される思考形態に極めて類似している。彼らサポート担当者が電話でプリンタ問題を処理する時、彼らは、上記と同様な原因および副次的原因のリストを念頭に置き、顧客との会話に基づいて様々な原因の「信念(belief)」を絶えず調節する。

【0055】ステップ902において、もし存在するならば、副次的原因が識別される。原因をカテゴリに分類することが便利であることが多い。これらのカテゴリは、多数の副次的原因を含む原因とみなされる。すべての副次的原因を同じトップ・レベルに持たせることが可能であるので、原因の副次的原因を使用することは必ずしも

必要ではない。しかしながら、このような手法は、非常に多数の原因がトップ・レベルに至ることが多くなるので、確率を取得するのが一層困難になる。原因を階層に構成することは、領域専門家が、確率を評価する時に一度に比較的少ない原因を考慮することを可能にし、その結果得られる情報の精度が上がる。

【0056】図4には、2つのレベルの原因構造のみが表されているが、任意の多くのレベルの原因および副次的原因を利用することができる。“薄い印刷”の原因の最終的階層が次の表4に示されている。

【0057】

【表4】

1) 用紙
2) 汚れた用紙経路
3) 環境条件
4) トナー・カートリッジ問題 <ul style="list-style-type: none"> <li>a) 欠陥のあるトナー・カートリッジ</li> <li>b) 正しく取り付けられてないトナー・カートリッジ</li> <li>c) トナー分散 - これは、トナーが足りない、およびトナー流体に関するその他の問題を含む。</li> </ul>
5) 転写ローラ問題 <ul style="list-style-type: none"> <li>a) 欠陥または汚れた転写ローラ</li> <li>b) 正しく取り付けられてない転写ローラ</li> <li>c) 摩耗した転写ローラ</li> </ul>
6) アプリケーションの不正確な設定 <ul style="list-style-type: none"> <li>a) 節約（エコノ）モード／ドラフト・モードのオン - 節約モードはトナーを節約するために設定されるものであり、そのため通常より薄い印刷の原因となる。</li> <li>b) 300/600dpiを300dpiに設定 - 300dpiは600dpi印刷より薄い印刷を引き起こすことがある。</li> <li>c) その他の間違った設定 - 薄い印刷を引き起こす可能性のあるその他の設定</li> </ul>
7) プリンタ・ドライバの不正確な設定 <ul style="list-style-type: none"> <li>a) 節約モードをオンに設定</li> <li>b) 300/600dpiを300dpiに設定</li> <li>c) その他の間違った設定</li> </ul>
8) コントロール・パネルの不正確な設定 <ul style="list-style-type: none"> <li>a) 節約モード／ドラフト・モードのオン</li> <li>b) 300/600dpiを300dpiに設定</li> <li>c) 印刷密度の低すぎる設定</li> </ul>
9) 損なわれたデータフロー
10) 間違ったドライバの使用
11) その他の問題 <ul style="list-style-type: none"> <li>a) 一時的問題</li> <li>b) 永久的問題</li> </ul>

【0058】ステップ903において、その問題のトラブルシューティング・ステップ（手順）が識別される。問題の原因のいずれかを解決することができるアクション、およびそれら原因に関する情報を提供することができる質問のリストが作成される。

【0059】問題のトラブルシューティング・ステップのリストを作成する時、領域専門家は、基本的には、彼らがその問題に直面した場合に彼ら自身が実行するまたは顧客に提案するステップを考察する。ともすれば手順を忘れがちであるので、すでにリストされている原因を考慮せずに（すなわち白紙の状態から）、ステップをリストにすることから開始するのが有益であることは、経験が示している。その後、そのような第1のステップがリストされたならば、原因のリストを検討して、それらの原因を潜在的に解決するすべてのステップを追加するのが良い。

【0060】トラブルシューティングのステップをリストにする時、トラブルシューティング機構の想定した使用者によって実行されることのできるステップだけをリ

ストすべきである。すなわち、例えば、使用者がエンドユーザであれば、実行の成功には印刷システムの高い技術的理解が要求されるようなステップを提案することは適切ではない。また、未熟なユーザによって実行される場合には何かを壊すリスクが高いようなステップもあるが、それは含めるべきではない。非常に高価な必需品を必要とするステップもまた、通常含めるべきではない。

【0061】この場合もまた、領域専門家は、ステップのサイズおよび範囲という問題に直面する。単一のステップまたは一連のステップとして同等にモデル化されることができるトラブルシューティングのプロシージャが存在する。それが、ユーザ・インタフェース、およびステップを表現するステップ自体に依存することは経験から明らかである。もしもステップが、決定性流れ図のif-then-else構文としてうまく表現されることが可能であり、またトラブルシューティング機構のユーザ・インタフェースがそのような決定性"プログラム"の実現を支持するとすれば、そのステップは単一ステップとしてモデル化されるべきであろう。ステップの流れ図が不確実性

/確率的判断を含むとすれば、そのステップは、複数のステップとして表現されるべきであろう。

【0062】トラブルシューティングのステップ、アクションおよび質問には2つの主要なカテゴリがある。第1のカテゴリはアクションであり、ユーザが、システムにおいて何らかの介入を実行し、そのアクションが問題を解決したか否かに関してトラブルシューティング機構へ報告を返すことを要求するステップである。このように、アクションは問題を解決する潜在性を持つ。第2のカテゴリは質問であって、ユーザが、システムへの介入によって問題に関係する何らかの情報を手元に入手して、結果をトラブルシューティング機構に報告を返すことを要求するステップである。質問は、情報収集アクションと一般的質問という2つのサブカテゴリに分類される。

【0063】情報収集アクションは、問題を解決する潜在性を持たないアクションである。これは、問題を解決することに関連している情報を単に提供するだけである。普通のアクションは解決アクションとも呼ばれ、情報収集アクションとは区別される。トラブルシューティング・アルゴリズムにおける2つのタイプのアクションの取り扱いが異なるので、情報収集アクションが質問として取り扱われる以下の記述において、それらを区別することは重要である。これは、アルゴリズムとして情報収集アクションと質問との間に相違がないことを意味する。しかし、領域専門家が、情報収集アクションをアクションとみなせば、それら情報収集アクションの確率を一層容易に導出することができるので、知識獲得の間は

その区別が維持される。

【0064】情報収集アクションと解決アクションとの間の相違もまた明らかにされなければならない。解決アクションは問題を解決する潜在性を持つが、情報収集アクションは問題を解決する可能性を持たない。情報収集アクションは、環境への何らかの変更が試されるとはいえ、一時的に問題を取り除く潜在性を持つにすぎない。

【0065】一般的質問は、情報収集アクションではない残りの質問である。質問は、問題を解決する潜在性を持たず、YES(役立った)とNO(役立たなかった)という2つの回答を持つにすぎないアクションとは対照的に、任意の数の回答を持つことができる。

【0066】問題のトラブルシューティング・ステップのリストを作成する時、それらステップは、解決アクション(solution actionを略してSAと呼称される)、情報収集アクション(information-gathering actionを略してIAと呼称される)、または質問(questionを略してQと呼称される)として分類されなければならない。

【0067】すべてのアクションおよび質問に関して、知識獲得プロセスにおける可能な限り早い段階において、説明を書き出されなければならない。なぜならば、そのような説明/定義は将来の混乱を減少させ、エラーができるだけ早期に捕捉されることを確実にする上で役立つからである。

【0068】"薄い印刷"問題に関して、次の表5に示されるようなステップが特定される。

【0069】

【表5】

A) 用紙が仕様に含まれていることを確認する (SA)
B) 仕様に含まれている別のトナー・カートリッジを試す (IA)
C) トナー・カートリッジを取り出し、振って、元通りに差し込む (SA)
D) 転写ローラの取り付けを再度行う (SA)
E) 異なる用紙を試す (IA)
F) プリンタ保守キットを実行する (SA)
G) プリンタの電源を一旦切って、再度入れる (SA)
H) 環境条件が仕様の範囲内にあることを確認する (SA)
I) ユーザ・マニュアルに従ってプリンタの内部を掃除する (SA)
J) 仕様に準拠した別の転写ローラを試す (IA)
K) アプリケーションにおいて節約／ドラフト・モードがオンになっていないことを確認する (SA)
L) アプリケーションにおいて300dpiが設定されていないことを確認する (SA)
M) "薄い印刷"に関連するその他のアプリケーション設定を調べて訂正する (SA)
N) プリンタ・ドライバにおいて節約モードがオンになっていないことを確認する (SA)
O) プリンタ・ドライバにおいて300 dpiが設定されていないことを確認する (SA)
P) "薄い印刷"に関連する他のプリンタ・ドライバ設定を調べて訂正する (SA)
Q) プリンタのコントロール・パネル上で節約／ドラフト・モードがオンになっていないことを確認する (SA)
R) プリンタのコントロール・パネル上で300dpiが設定されていないことを確認する (SA)
S) コントロール・パネル上で印刷密度の設定が低すぎないことを確認する (SA)
T) データフローのトラブルシューティングを行う (SA)
U) 仕様に準拠した、最新のプリンタ・ドライバが使用されていることを確認する (SA)
V) プリンタ保守キットは作動時期になっているか？ (Q)
W) トナー・カートリッジは正規の製造業者からのものか？ (Q)
X) "トナーが少ない"というメッセージがコントロール・パネルに表示されていないか？ (Q)
Y) 印刷構成ページが薄く印刷されているか？ (Q)

【0070】上記ステップのうち、たとえば"B)仕様に含まれている別のトナー・カートリッジを試す (IA)"のようないくつかのステップは、情報収集アクションとして分類されている。ステップBを実行した後に問題が取り除かれたとしても、問題はまだ解決されていない。その問題についてありそうな原因は識別されたが、更なる調査を実行して、他のトナー・カートリッジを元の場所に戻さなければならぬかもしれない。つまり問題は解決していない。これは、プリンタ・コンポーネントを他のものと取り替えるステップについて一般的にあてはまる。すなわち、取り替えが成功しても、トラブルシューティングの範囲は大幅にしばられたが、問題を完全に解決するために実行することができるステップがなお残っている。

【0071】表5のステップFは、特定の量のページが印刷される都度、プリンタ保守キットが実行されなけれ

ばならないことを提示している。プリンタ保守キット (PMキット) が実行されなければならない場合には、プリンタのコントロール・パネルが通常それを通知するが、必ずしも常に通知するとは限らない。プリンタ保守キットは絶対に必要である場合にのみ実行されるべきであるので、プリンタ保守キットを提示する前に、コントロール・パネル上でそれを提示するか否かを尋ねるようになるのが良い。

【0072】表5のステップTは、印刷ジョブがデータフローにおいて損なわれているか否かを判断し、その損なわれた源の識別を試みる一連のサブステップを含む、大きくて複雑なトラブルシューティング・ステップである。基本的には、以下に記述される、損なわれた出力に関するデータフロー・モデル全体が、ステップTおよびその関連する原因の下に適合される。

【0073】ステップ904において、原因およびトラブ

ルシューティング・ステップが対応づけられる。トラブルシューティング・ステップは、それらが解決することができる原因と対応づけされる。加えて、質問に関連づけられた原因が識別される。この手順において、アクションは、それが解決することのできる原因と対応づけられ、質問は、それが関連づけられた(すなわち、その確率に影響を及ぼす)原因と対応づけられ、こうして原因が、トラブルシューティング・ステップに対応づけられる。

【0074】各アクション $A_i$ について、各原因 $C_j$ に関して $A_i$ の実行が $C_j$ を解決するゼロでない確率があるかどうか考察される。あるとすれば、対応があるとして、知識獲得プロセスにおいて後で使用するために登録される。

【0075】情報収集アクションは、解決アクションとほとんど同じように取り扱われることができる。情報収集アクションは、問題を解決することができないとしても、環境におけるなんらかの変更を試みながら一時的に問題を取り除くことができる。例えば、上記表5における"仕様に含まれている別のトナー・カートリッジを試す"というステップbは、その原因が表4にリストされている副次的原因4a、4bまたは4cであるならば、その問題を取り除く。従って、情報収集アクションに関しては、

実行される時に問題を除去するであろうアクションに対する原因が登録される。

【0076】各質問 $Q_i$ について、各原因 $C_j$ に関して $Q_i$ に対する回答が、 $C_j$ の信念に直接影響するか否か(すなわち、確率を減少または増加させる結果を生むか否か)が考察される。

【0077】質問は、必ずしも原因の信念に影響を及ぼす必要はなく、トラブルシューティング・シナリオやユーザの種類等に関する情報を提供して関連するアクションを可能または不可能にするために、質問が使用されることが時としてある。この場合の例をあげると、特定のコンポーネントの種類または製造者に関する質問であり、その質問に対する回答により、特定のアクションをそのコンポーネントがサポートするか否かが制御される。このように、コンポーネントの製造者が正しい種類のものでないとき、これらのアクションが成功する確率はゼロである。

【0078】"薄い印刷"問題に関して、ステップと原因の対応付けが次の表6に示されている。各アクションまたは質問の後に、(表4における参照符号を使用して)関連する原因がリストされている。

【0079】

【表6】

トラブルシューティング・ステップ	原因
A) 用紙が仕様に含まれていることを確認する (SA)	1
B) 仕様に含まれている別のトナー・カートリッジを試す (IA)	4
C) トナー・カートリッジを取り出し、振って、元通りに差し込む (SA)	4 b、4 c
D) 転写ローラの取り付けを再度行う (SA)	5 b、 1 1 a
E) 異なる用紙を試す (IA)	1
F) プリンタ保守キットを実行する (SA)	2、5、 1 1 a
G) プリンタの電源を一旦切って、再度入れる (SA)	1 1 a
H) 環境条件が仕様の範囲内にあることを確認する (SA)	3
I) ユーザ・マニュアルに従ってプリンタの内部を掃除する (SA)	2、4 b、 1 1 a
J) 仕様に準拠した別の適合転写ローラを試す (IA)	5、1 1 a
K) アプリケーションにおいて節約／ドラフトモードがオンになっていないことを確認する (SA)	6 a
L) アプリケーションにおいて300dpiが設定されていないことを確認する (SA)	6 b
M) "薄い印刷"に関連する他のアプリケーション設定を調べて訂正する (SA)	6 c
N) プリンタ・ドライバにおいて節約モードがオンになっていないことを確認する (SA)	7 a
O) プリンタ・ドライバにおいて300 dpiが設定されていないことを確認する (SA)	7 b
P) "薄い印刷"に関連する他のプリンタ・ドライバ設定を調べて訂正する (SA)	7 c
Q) プリンタのコントロール・パネル上で節約／原稿モードがオンになっていないことを確認する (SA)	8 a
R) プリンタのコントロール・パネル上で300dpiが設定されていないことを確認する (SA)	8 b
S) コントロール・パネル上で印刷密度の設定が低すぎないことを確認する (SA)	8 d
T) データフローのトラブルシューティングを行う (SA)	9
U) 仕様に準拠した、最新のプリンタ・ドライバが使用されていることを確認する (SA)	1 3
V) プリンタ保守キットは作動時期になっているか？ (Q)	2、5、 5 c
W) トナー・カートリッジは正規の製造業者からのものか？ (Q)	4
X) "トナーが少ない"というメッセージがコントロール・パネルに表示されていないか？ (Q)	4、4 c
Y) 印刷構成ページが薄く印刷されているか？ (Q)	1-5、 8、1 1

【0080】表6において、トラブルシューティング・ステップvは、領域専門家によると、原因2、5および5cの信念に影響を及ぼす。プリンタ保守キットが作動時期になっているならば、プリンタ保守キットによって対象とされる原因のいくつか、すなわち、(2)汚れた用紙経路、(5)一般的な転写ローラの問題、および(5c)転写ローラの摩耗という原因に対する信念が比較的高い。

【0081】トラブルシューティング・ステップyにおける質問は、構成ページが薄く印刷されているか否かと 50

いう徴候に関する情報を要求する。これは、原因1-5、8および11の徴候である。これらの原因は、構成ページが印刷される時にも影響を持つハードウェア原因である。ここで規定されていない原因は、この状況では影響を持たないソフトウェア原因である。質問に関する確率の取得については、後述される。

【0082】ステップ905において、新しい原因または副次的原因が識別されたか否か検証される。これは、例えば原因とステップが対応づけられる時に識別されるこ

とができる。なんらかの新しい原因または副次的原因が識別されると、ステップ901に戻る。

【0083】アクションおよび質問を、それらに関連する原因と対応づける時、解決アクションの無い原因がしばしば発見され、またいずれの原因をも解決することができないアクションが発見される。すなわち、アクションが欠けているもの、および原因が欠けているものがある。このような場合、ステップ901に戻るが必要とされる。

【0084】ステップ906において、例えば原因と手順を対応づける時、新しいトラブルシューティング・ステップが識別されたか否か検証される。新しいトラブルシューティング・ステップが識別されたならば、ステップ903へ進む。

【0085】初期のリスト作成において原因およびステップがしばしば忘れられ、原因をステップに対応づける時、新しい原因およびステップが識別されることが多い。従って、新しい原因が発見されるたびに確率の導出を部分的にもう一度実行しなければならないので、原因についての確率を導出する前に原因とステップの対応づけを実行することが最適である。

【0086】ステップ907において、原因および副次的原因の確率が評価される。すべての原因のリストが作成され、原因および副次的原因が階層的に構成されて確実性が高くなった時、原因の確率を評価すべきである。これは、通常、原因を所与とした場合の副次的原因の確率が先ず評価され、次に、問題を所与とした場合の原因の確率が評価されるというように、下から上へと行われる。

【0087】副次的原因の確率が最初に評価される。副次的原因セットは、確率の別個の導出が、同じ原因のそれぞれの副次的原因セットに対して実行されるような順序で処理される。問題(例えば「薄い印刷」)が存在し、その原因(例えば「トナー・カートリッジ問題」)が存在すると仮定して、副次的原因の確率が導出される。副次的原因のすべての確率が導出された時、問題が存在すると仮定して、原因の確率が導出される。

【0088】確率が基本的に(副次的原因が原因を引き起こし、原因が問題を引き起こすという)因果関係の方向に対して導出される場合、領域専門家は、問題および/または原因が存在すると仮定することが許され、領域専門家に確率の基礎となる最大限の情報を付与することができるので、確率導出のこの方法は非常に効率的であることが経験によって示されている。

【0089】原因/副次的原因のセットの確率を導出する通常のプロシージャは、一人の領域専門家が、より高いレベルの原因、または少なくともランキング(これは最も高い、これは次に最も高い、等)が与えられた

場合に、ほとんどの原因に初期の確率を与えるものである。その後、領域専門家は、初期確率またはランキングを考察して、その結果調整を行う。最終的合意に達すると、導出が終了する。

【0090】導出プロセスにおいて起こる信念の相違は、領域専門家のうちの一人による知識不足によるものがほとんどであり、そのため、領域専門家のだれが間違っているかを発見する議論が起きる。ほとんどの場合、迅速に合意に達し、その合意を反映するように確率が調整される。しかし、時として、意見の相違を解決するため他の専門家と相談することが必要とされる。

【0091】確率における意見の相違が非常に小さいとき(例えば0.05)、長時間の議論は不必要であるとみなされ、平均が選択される場合が多い。しかしながら、意見の相違が大きい時、根本的な領域構造の共通の理解に達することは、将来の確率導出の観点から非常に重要である。

【0092】導出プロセスの間、検討中の原因に対して1つの確率セットが展開される。この確率セットは、必ずしも(和が1.0となるように)正規化される必要はない。和をわずかに1.0と異なるように柔軟にしてはならない理由はない。常に和が1.0に維持されなければならないとすれば、プロセスは大幅に遅くなる。導出が終了した後、確率を正規化するのは簡単である。

【0093】プロジェクトによっては、領域専門家達が、0.1などのような確率ではなく10%のようなパーセンテージで導出することを望むことがある。0から1の範囲に比較して、0から100の範囲の方が小数が少ないので作業しやすく、これは意味を持つ。また、彼らがパーセントで思考するのに慣れているということは、ありそうなことである。

【0094】明らかに、導出された確率にながしかの量の二次的不確実性が常に存在する。この二次的不確実性を表現する1つの標準的方法は、確率間隔を使用するものである。この場合、領域専門家は、確率がある特定の区間の範囲内にあるという信念を記述する。領域専門家が特定の区間について同意した時、確率区間のページアン・ネットワークへの伝播を可能にする方法がある。二次的不確実性を明示的にすることによって、領域専門家は、異なる確率について異なるサイズの確率区間を規定することが可能となり、自動化されたトラブルシューティング機構が、適切な不確実度をもってその結論を出すことができるようになる。

【0095】「薄い印刷」問題に関して、次の表7に示されるような確率が導出される。

【0096】

【表7】



1) 用紙	1 2	
2) 汚れた用紙経路	2 2	
3) 環境条件	6	
4) トナー・カートリッジ問題 a) 欠陥のあるトナー・カートリッジ b) 正しく取り付けられてないトナー・カートリッジ c) トナー分散	3 5	1 0 5 8 5
5) 転写ローラ問題 a) 欠陥または汚れた転写ローラ b) 正しく取り付けられてない転写ローラ c) 摩耗した転写ローラ	2 0	2 5 6 5 1 0
6) 不正確なアプリケーション設定 a) 節約モード/ドラフト・モードのオン b) 300/600dpiを300dpiに設定 c) その他間違った設定	4 5	7 5 1 5 1 0
7) 不正確なプリンタ・ドライバ設定 a) 節約モードをオンに設定 b) 300/600dpiを300dpiに設定 c) その他間違った設定	2 7	8 0 1 5 5
8) 不正確なコントロール・パネル設定 a) 節約モード/ドラフト・モードのオン b) 300/600dpiを300dpiに設定 c) 印刷密度の低すぎる設定	2	8 5 5 1 0
9) 損なわれたデータフロー	1	
10) 間違ったドライバの使用	1 5	
11) その他の問題 a) 一時的問題 b) 永久的問題	1 0	1 0 9 0

【0097】ステップ908において、アクションおよび質問の確率が評価される。本発明の好ましい実施形態において、原因の徴候または影響に付随する質問と、性質上徴候または影響とみなされない一般的質問という2つのタイプの質問がある。これら2つのタイプの質問に関する知識獲得プロセスは異なるので、確率を導出する前に質問のタイプを決定することは重要である。これら2つのタイプの質問の間の相違について以下記述する。

【0098】一般的質問に関しては、質問に関連する原因、すなわち質問に対する回答に依存して確率が増減する原因のリストがすでに作成されている。このタイプの質問に関しては、領域専門家は、質問に対する (YES、NO 等の) 回答のそれぞれを考慮し、新しい情報に基づいて、影響を受けた原因の確率がどれだけ増減するか評価する。導出は、原因の場合とほとんど同じように進められる。すなわち、意見が合わない点は議論によって解決されなければならない。

【0099】領域専門家は、質問に対する回答によって

影響を受ける原因に重点をおく。従って、影響を受けない原因の確率は、専門家によって修正されない。しかしながら、他の確率がそれぞれ増減された確率を持つという事実により、それに応じて残りの確率が、合計がなお1.0となるように変更される。専門家がすべての確率の変更を評価するのではなく、直接影響を受ける確率だけを調整し、残りはそれに応じて変更するという方法の方が容易であることは明らかである。また、経験によれば、直接影響を受ける確率の調整に応じて残りの確率を適切に変更する方が、専門家にとっても望ましく感じる。

【0100】"薄い印刷"問題において、"コントロール・パネル上にトナーが少ないという標示が見えるか?"という質問に対する回答があった場合、確率(パーセンテージで表現)は、以下の表8に記述されるように調整される。

【0101】

【表8】

			Y e s	N o
1) 用紙	1 2			
2) 汚れた用紙経路	2 2			
3) 環境条件	6			
4) トナー・カートリッジ問題 a) 欠陥のあるトナー・カートリッジ b) 正しく取り付けられてないトナー・カートリッジ c) トナー分散	3 5	1 0 5 8 5	-> 9 0	-> 2 5
5) 転写ローラ問題 a) 欠陥または汚れた転写ローラ b) 正しく取り付けられてない転写ローラ c) 摩耗した転写ローラ	2 0	2 5 6 5 1 0		
6) 不正確なアプリケーション設定 a) 節約モード/ドラフト・モードのオン b) 300/600dpiを300dpiに設定 c) その他間違った設定	4 5	7 5 1 5 1 0		
7) 不正確なプリンタ・ドライバ設定 a) 節約モードをオンに設定 b) 300/600dpiを300dpiに設定 c) その他間違った設定	2 7	8 0 1 5 5		
8) 不正確なコントロール・パネル設定 a) 節約モード/ドラフト・モードのオン b) 300/600dpiを300dpiに設定 c) 印刷密度の低すぎる設定	2	8 5 5 1 0		
9) 損なわれたデータフロー	1			
10) 間違ったドライバの使用	1 5			
11) その他の問題 a) 一時的問題 b) 永久的問題	1 0	1 0 9 0		

【0102】このように、プリンタのコントロール・パネルがトナーが少ないことを標示する場合、問題の原因である「トナー・カートリッジ問題」の確率が0.9に増加されている。「トナー分散」という副次的原因の確率は「トナー・カートリッジ問題」のその他の副次的原因に比較してすでに高くなっているため、この確率を更に増加させないことが決定されている。

【0103】同じように、コントロール・パネルがトナーの少ないことを標示していないことを前提として、「トナー分散」という副次的原因の確率は、0.85から0.25へ減らされている。しかしながら、たとえコントロール・パネルがトナーの少ないことを標示していないことを前提としても、「トナー・カートリッジ問題」の全体的確率は0.35のままとすることが決定されている。

【0104】また、一般的質問に関して、領域専門家は、質問に対する回答に事前確率を与えなければならない。詳細は後述するが、専門家が一般的質問に関して矛盾した情報を指定したか否かは、関連する原因の無条件

確率 $P(C)$ 、条件つき確率 $P(C|Q)$ および質問に対する事前確率 $P(Q)$ を分析することによって、すなわち、 $\sum Q P(C|Q) P(Q)$ を $P(C)$ と比較することによって、検証される。

【0105】徴候についての質問に関しては、質問に関連づけられた原因、すなわち対象となっている徴候を引き起こす原因が、上述の図4のステップ904においてリストされる。ここでは、関連する原因のそれぞれについて、原因を所与とした場合の徴候の確率を与えることにより導出が行われる。また、指定された原因のいずれもが存在しない場合に徴候が出現するという確率を評価すべきである。

【0106】「薄い印刷」問題において、「構成ページの印刷は薄いか？」という質問(表5の質問v)は徴候質問である。確率(パーセンテージ表現)は、次の表9のように評価される。

【0107】

【表9】

	Y e s
1) 用紙	1 0 0
2) 汚れた用紙経路	1 0 0
3) 環境条件	1 0 0
4) トナー・カートリッジ問題 a) 欠陥のあるトナー・カートリッジ b) 正しく取り付けられてないトナー・カートリッジ c) トナー分散	1 0 0
5) 転写ローラ問題 a) 欠陥または汚れた転写ローラ b) 正しく取り付けられてない転写ローラ c) 摩耗した転写ローラ	1 0 0
6) 不正確なアプリケーション設定 a) 節約モード/ドラフト・モードのオン b) 300/600dpiを300dpiに設定 c) その他間違った設定	
7) 不正確なプリンタ・ドライバ設定 a) 節約モードをオンに設定 b) 300/600dpiを300dpiに設定 c) その他間違った設定	
8) 不正確なコントロール・パネル設定 a) 節約モード/ドラフト・モードのオン b) 300/600dpiを300dpiに設定 c) 印刷密度の低すぎる設定	1 0 0
9) 損なわれたデータフロー	
10) 間違ったドライバの使用	
11) その他の問題 a) 一時的問題 b) 永久的問題	5 0

【0108】指定された原因が全く無い場合、徴候の確率（パーセンテージとして）は1である。

【0109】このように、領域専門家の評価によれば、例えば、原因が不正確なコントロール・パネル設定（上の表9における原因8）であれば、構成ページが薄く印刷される確率は1.0(100%)であり、また、原因が用紙、用紙経路、環境条件等のいずれかであった場合も同様である。原因が"その他の問題"であれば、構成ページが薄く印刷される確率は0.5と専門家は評価している。この確率が1.0でない理由は、一時的および永久的問題のいくつかは構成ページを印刷することに影響を及ぼさないことにある。

【0110】たとえ上記の指定された原因のいずれも存在しなかったとしても、領域専門家は、構成ページが薄く印刷されるという可能性を完全に除外しようとはしない。従って、そのような状況についての確率は、0.01(1%)としている。

【0111】図7に示されているデータフロー・モデルには、質問の第2のカテゴリであるグローバル質問がある。グローバル質問は、一般的質問として、情報収集または標準的質問の両方でありうる。これらは、普通の質問より高いレベルで作用するのでグローバルと言われ、

データフローにおけるどのコンポーネントが問題を引き起こしたかを求めるのに使用される。グローバル質問に関する知識獲得は、通常の質問の場合と同様に行われるが、原因がコンポーネントによって置き換えられる。

【0112】アクションに関しては、図4のステップ904にリストされた原因のそれぞれが与えられた場合の、アクションが問題を解決する確率を決定することが必要である。これらの原因は、アクションが潜在的に解決することができる原因であると仮定される。

【0113】トラブルシューティング・アルゴリズムは、問題に関してすでに得られた情報が与えられた場合の、その問題を解決するアクションの確率を必要とする。従って、領域専門家は、リストされた原因のそれぞれCiに関して、対象となっているCiが問題の唯一の原因であると仮定して、そのアクションを実行することがその問題を解決する確率はいくらであるか答えなければならない。

【0114】経験によれば、この確率を評価する時、アクションが正しく実行されたとした場合のそのアクションが問題を解決する確率のみならず、そのアクションが正しく実行される確率など非常に多くの事柄を考慮しなければならない。あまりに多くの事柄が同時に考慮され

30

40

50

なければならないとすれば、確率の品質が低下する結果をもたらす。

【0115】上記の導出が2つの確率導出質問に分割されれば、評価の品質は高くなる。第1の確率導出質問は、対象となっているC<sub>i</sub>が問題の唯一の原因であると仮定して、そのアクションを正しく実行することがその問題を解決する確率がいくらであるか、というものである。第2の確率導出質問は、対象となっているC<sub>i</sub>が問題の唯一の原因であると仮定して、ユーザが気づかずに間違っ

てアクションを実行する確率がいくらであるか、というものである。

【0116】第1の確率導出質問に答える時、領域専門家は、アクションが正しく実行されたと仮定することができるので、アクションが問題を解決する確率を評価することは比較的容易である。第2の確率導出質問に答える時、領域専門家は、ユーザが間違っ

てアクションを実行する確率を評価することが重要であって、アクションを間違っ

て実行する全般的確率を評価することが重要なのではない。この確率は、ユーザからの不正確なフィードバックの可能性を表すのに必要である。ユーザが間違っ

てアクションをしたことを自覚しない状況において、不正確なフィードバックが得られる。従って、ユーザが間違っ

てアクションをしたことを自覚する状況は、確率に含まれない。そのような状況においては、ユーザは不正確なフィードバックを入力せず、アクションを再度実行することを試みるか、またはアクションを実行することができなかつたことを入力として与えるであろう。

【0118】第1の確率導出質問に答える時に見出される確率がP<sub>1</sub>で表され、第2の確率導出質問に答える時に見出される確率がP<sub>2</sub>で表されるとすれば、原因C<sub>i</sub>を所与として、アクションが問題を解決する全般的確率は、 $P(A=yes | C_i=yes) = P_1(1 - P_2)$ である。経験によれば、第2の確率導出質問(ユーザ応答の不確実度とも呼ばれる)に答える時に評価される確率には、変動がほとんど見られない。従って、0-非常に低い、1-低い、2-中位、3-高い、4-非常に高いというレンジを使用して不確実度について0から4の間の因子を評価すれば、十分

【0119】

【表10】

V L (非常に低い)	0
L (低い)	2 %
M (中位)	5 %
H (高い)	10 %
V H (非常に高い)	20 %

【0120】アクションの確率を評価する時に設定されるべき仮定が更にいくつかある。すなわち、アクションを実行するために必要な前提条件があれば、アクションが提案される時にそれら条件が利用可能にあると仮定される。従って、アクションが問題を解決する確率を評価する時、必要条件の利用可能性を考慮する必要はない。必要条件の利用可能性は、ユーザがアクションを実行することができないことまたは実行することを望まないことを報告することによりユーザにそのアクションを飛ばさせる(スキップさせる)ことを可能にすることにより制御される。

【0121】アクションが、疑わしいコンポーネントを他のものと取り替えることを伴う時、新しいコンポーネントが不完全で同じ問題を引き起こす可能性がわずかにある。たとえこの確率が無視できるほどのものであることが多いとしても、アクションが問題を解決する確率を評価する時、この確率を考慮に入れることが必要である。代替コンポーネントが不完全で同じ問題を引き起こすならば、ユーザは、アクションが役立たなかったことをトラブルシューティング・システムに入力する。システムは、代替コンポーネントが不完全な可能性があるからには、アクションが解決することができる原因を完全に取り除くべきではない。

【0122】上述のように、解決アクションと情報収集アクションの間には相違がある。たとえば情報収集アクションが問題を解決することができないとしても、確率は全く同様に収集される。実際、たとえば情報収集アクションが問題を解決することができないとしても、それらアクションは、構成が変わった時に問題がなくなるか否かを調べる検査をシステム上で実行する。その後、対象となっているC<sub>i</sub>が問題の唯一の原因であると仮定して、新しい構成においてアクションを正しく実行することにより問題を解決する確率がどれほどのものか?というように、上述の第1の確率導出質問が若干変更される。

【0123】"薄い印刷"問題に関して、アクションの確率は次の表11のように評価される。それぞれのアクションの後に、関連する原因およびアクションが原因を解決する確率のリストが作成される。不確実因子については、後述する。

【0124】

【表11】

アクション	原因:確率
A) 用紙が仕様に含まれていることを確認する (SA)	1:100
B) 仕様に含まれている別のトナー・カートリッジを試す (IA)	4:100
C) トナー・カートリッジを取り出し、振って、元通りに差し込む (SA)	4b:100, 4c:100
D) 転写ローラの取り付けを再度行う (SA)	5b:100, 11a:100
E) 異なる用紙を試す (IA)	1:90
F) プリンタ保守キットを実行する (SA)	2:100, 5:100, 11a:100
G) プリンタの電源を一旦切って、再度入れる (SA)	11a:100
H) 環境条件が仕様の範囲内にあることを確認する (SA)	3:100
I) ユーザ・マニュアルに従ってプリンタの内部を掃除する (SA)	2:100, 4b:100, 11a:100
J) 仕様に準拠した別の転写ローラを試す (IA)	5:100, 11a:100
K) アプリケーションにおいて節約/ドラフトモードがオンになっていないことを確認する (SA)	6a:100
L) アプリケーションにおいて300dpiが設定されていないことを確認する (SA)	6b:100
M) "薄い印刷"に関連する他のアプリケーション設定を調べて訂正する (SA)	6c:100
N) プリンタ・ドライバにおいて節約モードがオンになっていないことを確認する (SA)	7a:100
O) プリンタ・ドライバにおいて300 dpiが設定されていないことを確認する (SA)	7b:100
P) "薄い印刷"に関連する他のプリンタ・ドライバ設定を調べて訂正する (SA)	7c:100
Q) プリンタのコントロール・パネル上で節約/原稿モードがオンになっていないことを確認する (SA)	8a:100
R) プリンタのコントロール・パネル上で300dpiが設定されていないことを確認する (SA)	8b:100
S) コントロール・パネル上で印刷密度の設定が低すぎないことを確認する (SA)	8d:100
T) データフローのトラブルシューティングを行う (SA)	9:100
U) 仕様に準拠した、最新のプリンタ・ドライバが使用されていることを確認する (SA)	13:100

【0125】ステップ909において、アクションおよび質問のコストが評価される。トラブルシューティング・アルゴリズムにおいて、次に実行する最適のステップがどれであるかを判断することができるようにするため、アクションおよび質問を実行するコストを知ることが必要である。コストは、単一の因子、または複数の因子の組合せのいずれとしてでも評価することができる。コストは、実際には複数の重要な因子から構成されるので、これらの因子のそれぞれを別々に評価し、次にそれら因子を単一のコスト因子に結合することが、最も信頼の高い正確な手法と思われる。コストは多くの因子から

40 構成される。次の4つの因子が最も重要とみなされる。

【0126】第1の因子は時間である。時間は1つのステップを実行するために要する時間(分で測定される)である。労働に費やされる時間は、待つことに費やされる時間と区別される。待ち時間の重さは労働時間より小さい。これは、大部分が待ち時間である10分を要するステップは、常に労働する10分を要するステップより低いコストを与えられることを意味する。時間を評価する時、時間はユーザ母集団に対して平均される。他のユーザより迅速に特定のステップを実行することができる経験豊かなユーザが存在するが、最終的な時間評価は、すべて

のタイプのユーザに対して平均されなければならない。

【0127】第2の因子はリスクである。リスクは、ステップを実行する時に何かを破壊するリスクであり、非常に低い、低い、中位、高い、非常に高いという5段階で評価される。ステップを提案する時、何かを壊す高いリスクのステップの前に最も低いリスクのステップを提案することが望ましいので、リスクは非常に意味がある。リスクもまた、何かを壊すリスクの低い経験豊かなユーザとリスクの高い初心者ユーザが含まれるユーザ母集団に対して平均されなければならない。

【0128】第3の因子は金額である。金額は、ステップの必要条件を購入するために必要なお金の量であり、非常に低い、低い、中位、高い、非常に高いという5段階で評価される。ステップによっては、ユーザがすべての必要条件を保持せず、それらを購入しなければならない可能性がある。そのようなステップは、必要条件を持たない類似したステップより高いコストを受けなければならない。この場合もまた、ステップのために必要な金額は、ユーザ母集団に対して平均されなければならない。ユーザのタイプ次第によって、その必要条件を持つかもしれないユーザもいれば、それらを購入しなければならないユーザもいる。

【0129】第4の因子は侮辱である。これは、ステップが提案されるときにユーザが感じる侮辱の度合いであり、非常に低い、低い、中位、高い、非常に高いという5段階で評価される。仮に経験豊かなユーザが(例えばプリンタの電気プラグが差し込まれているか調べて下さいというような)初歩の手順を提案されるとすれば、そのユーザは侮辱を感じるかもしれない。従って、侮辱が少な目のステップを一連のシーケンスの初めの段階で提案することを可能にするようなステップは、若干高いコストを与えられる。

【0130】ステップを実行する際の不便さというような、考慮することができるその他のコスト因子がいくつかある。しかしながら、経験に従えば、上記4つの因子だけで実際には十分である。ステップの不便さは、時間とリスクによって部分的に考慮に入れている(すなわち、ステップが不便であれば、それは一層長い時間を要し、一層高いリスクを含む)が、そのステップをスキップする能力によっても考慮される。

【0131】複数のコスト因子は、トラブルシューティング・アルゴリズムに役立つように1つの数値に結合されなければならない。これを実施するため、リスク、金額および侮辱因子が数値に変換され、最終的にバランスを取られ、加算されなければならない。これを実行する方法を求めるためには、領域専門家に、コスト因子の異

なるステップを分類する(ランクづけする)ように求める試みを何度も行うことが必要である。十分な回数のそのような試行を通して、変換の因子および加重を決定することができる。

【0132】専門家への質問の一例を示せば、時間=20でリスク=中位なA<sub>1</sub>、および、時間=10でリスク=高というA<sub>2</sub>のように、問題を解決する確率が等しい2つのアクションA<sub>1</sub>、A<sub>2</sub>のうちどちらを最初に提案することを望むか?というものである。印刷システム領域に関して、時間と互換性のある数値へのリスク因子の変換が次の表12に示される。

【0133】

【表12】

非常に低い	0
低い	1
中位	2
高い	4
非常に高い	8

【0134】この結果としての数値に9が乗じられる。すなわち、非常に高いリスクの0分ステップは、リスクの非常に低い72(8×9)分ステップに等しい。時間と互換性のある数値への金額因子の変換は、次の表13に示されている。

【0135】

【表13】

非常に低い	0
低い	1
中位	3
高い	10
非常に高い	30

【0136】この結果としての数値に10が乗じられる。すなわち、非常に高い金額因子の0分ステップは、金額の非常に低い300(30×10)分ステップに等しい。

【0137】侮辱因子は、印刷システム・プロジェクトにおいては、まれな状況においてのみ使用される。従って、完全な変換は定義されない。低い侮辱因子が指定される場合に、それが10に変換される。

【0138】"薄い印刷"問題に関して、不確実度およびコスト因子が次の表14に示されている。表の見出しにおけるI,T,R,MおよびIは、それぞれ順に、不確実度、時間、リスク、金額および侮辱を表している。

【0139】

【表14】

トラブルシューティング点順	I	T	R	M	I
A) 用紙が仕様に含まれていることを確認する (SA)	VH	15	0	0	0
B) 仕様に含まれている別のトナー・カートリッジを試す (IA)	L	7	L	0	0
C) トナー・カートリッジを取り出し、振って、元通りに差し込む (SA)	0	2	0	0	0
D) 転写ローラの取り付けを再度行う (SA)	M	4	L	0	0
E) 異なる用紙を試す (IA)	L	8	0	0	0
F) プリンタ保守キットを実行する (SA)	M	25	H	0	0
G) プリンタの電源を一旦切って、再度入れる (SA)	0	1	0	0	0
H) 環境条件が仕様の範囲内にあることを確認する (SA)	VH	120	0	0	0
I) ユーザ・マニュアルに従ってプリンタの内部を掃除する (SA)	L	7	L	0	0
J) 仕様に準拠した別の転写ローラを試す (IA)	L	1 0	M	0	0
K) アプリケーションにおいて節約/ドラフトモードがオンになっていないことを確認する (SA)	L	2	0	0	0
L) アプリケーションにおいて300dpiが設定されていないことを確認する (SA)	L	2	0	0	0
M) "薄い印刷"に関連する他のアプリケーション設定を調べて訂正する (SA)	L	5	0	0	0
N) プリンタ・ドライバにおいて節約モードがオンになっていないことを確認する (SA)	0	1	0	0	0
O) プリンタ・ドライバにおいて300 dpiが設定されていないことを確認する (SA)	0	1	0	0	0
P) "薄い印刷"に関連する他のプリンタ・ドライバ設定を調べて訂正する (SA)	L	5	0	0	0
Q) プリンタのコントロール・パネル上で節約/ドラフトモードがオンになっていないことを確認する (SA)	L	2	0	0	0
R) プリンタのコントロール・パネル上で300dpiが設定されていないことを確認する (SA)	L	2	0	0	0
S) コントロール・パネル上で印刷密度の設定が低すぎないことを確認する (SA)	L	2	0	0	0
T) データフローのトラブルシューティングを行う (SA)	VH	75	VH	0	0
U) 仕様に準拠した、最新のプリンタ・ドライバが使用されていることを確認する (SA)	L	15	L	0	0
V) プリンタ保守キットは作動時期になっているか? (Q)	0	1	0	0	0
W) トナー・カートリッジは正規の製造業者からのものか? (Q)	0	1	0	0	0
X) "トナーが少ない"というメッセージがコントロール・パネルに表示されていないか? (Q)	0	1	0	0	0
Y) 印刷構成ページが薄く印刷されているか? (Q)	0	4	0	0	0

【0140】ステップ910において、特別な制御を必要とするアクションおよび質問が識別され、処理される。

【0141】所望の通りに実行するトラブルシューティング機構をトラブルシューティング・モデルが取得するために、指定するのが必要な情報が更にいくつかある。それらは集散的に、特別な制御を必要とするアクションおよび質問と呼ばれる。

【0142】それらのうちの1つは「初期ステップ」で 50

ある。いくつかの問題について、後刻それらの調査を開始すると顧客が侮辱と感じるので初期的に除外されなければならないデフォルトの原因がある。例えば、「トレイ2上昇」というエラーコードの場合、トレイが上昇するのに十分な時間（これは、しばらく時間がかかることがあるので）をユーザが待たなかった可能性がある。従って、ユーザが十分な時間を待ったかどうかを最初に尋ね、待たなかったならばそうするように伝えるのが有益

である。このようなステップは常に最初に実行されるべきなので、これらをトラブルシューティング・ステップの通常の選択に含める理由はない。領域専門家はこのタイプのステップを識別して、それとわかるようにマークしなければならない。

【0143】指定すべきもう一つの情報は、「ワークアラウンド (workaround; 回避)」である。アクションは問題を解決することができるが、その解決が満足するものでないことがあるアクションは、そのことを示すワークアラウンドとして分類されることができる。例えば、メモリ不足という問題は、一層小さいジョブを印刷することによって問題を解決する。アクションがワークアラウンドとして分類されて、そのワークアラウンドが役に立った場合、ユーザは、解決に満足しているかどうか質問される。

【0144】指定すべき他の情報は、コンポーネの取り替えである。アクションがコンポーネントを他のものと取り替える場合には、コンポーネントが適切に取り付けられていない状況を自動トラブルシューティング機構が取り扱うことができるようにするため、これを登録することが重要である。コンポーネントを別のものと取り替える場合、そのコンポーネントが最初の位置に正しく取り付けられなかった可能性があるので、トラブルシューティング機構はこれを確認するため、ユーザに古いコンポーネントを再度取り付けてみるように促さなければならない。

【0145】指定すべき他の情報は、「取り消すことができないアクション」である。アクションが問題を解決しても原因が十分に識別されていなかったならば、ユーザは、トラブルシューティングの続行を望むかどうか尋ねられる。ユーザが続行に同意すれば、問題が再現するように、ユーザは最後のアクションを取り消さなければならない。(例えばPCリポート、プリンタの電源再投入のように)実行された最後のアクションが取り消せないものであれば、これは可能ではない。その状況においては、ユーザは、トラブルシューティングの続行を希望するか否か尋ねられるべきではない(それが可能ではないため)。従って、領域専門家は、取り消すことができないアクションを登録しなければならない。

【0146】指定すべき他の情報は、「含まれたアクション」である。アクションは、他のアクションを含むことがある。例えば、アクションが、プリンタの電力の切断と再投入のサイクルを含むことはよくあることで、そのようなアクションが実行された場合には、プリンタ電力切断再投入のサイクルをユーザに後刻再度提案すべきではない。従って、領域専門家は、アクションが他のアクションを含むか否かを登録しなければならない。

【0147】指定すべき他の情報は、「特例ステップ」である。特別の質問が特別の答えで回答された後のステップ、または特別の質問が特別の答えで回答されなかつ

た後のステップのように、特別のケースの場合にのみ提案されなければならないステップがある。例えば、印刷システム領域においては、コンポーネントの製造業者が検証された時のみ提案されるべき特別の製造業者固有のアクションがある。

【0148】指定すべき他の情報は「持続性」である。持続性は、後に実行されたアクションによって無効とされる、古い観察の問題を指す。例えば、質問QおよびアクションAが含まれる状況がしばしば存在するが、その場合にQがシステムの何らかのプロパティ(特性)の状態を要求し、その状態が所望のものでないとすれば、これを直すためアクションAが提案される。Qが所望の状態でないという観察のもとでは、トラブルシューティング機構が有効な情報について確実に動作するよう修正される。特別の状況でAの実行がQを解決するというような、質問およびアクションの対QおよびAが存在する状況を領域専門家に登録させることによって、この状況を取り扱うことができる。そのようにすれば、トラブルシューティング機構は、Qが前にどのように観察されたかにかかわらず、Aが実行されれば、その状況でQを自動的に解決する、ということを知る。これは、期待修理コスト(ECR)の計算に組み入れられていないので、なお完全な解決ではないということは明らかである。

【0149】図5は、自動顧客サポート運用印刷診断システムを示している(自動顧客サポート運用システムは、"a system for automated customer support operations"の頭文字を取って以下SACSOと呼ぶことがある)。SACSO印刷診断システムは、異なるタイプの印刷エラーをモデル化するいくつかのペイジアン・ネットワークから構成される。これらのネットワークおよびそれらの相互関係が図5に示されている。大きい円のそれぞれは、全モデルの1つのコンポーネントを表現する。

【0150】データフロー・モデル21は、顧客が印刷しようとした時にプリンタから出力を得られない場合、または、顧客が損なわれた出力を得る場合のすべてのエラーをカバーする。これらのエラーは、データを取り扱うアプリケーションからプリンタまでの流れ(これを、データフローと呼ぶ)におけるコンポーネントのいずれかによって引き起こされる可能性がある。

【0151】期待外出力モデル22は、プリンタに発生することがある期待外の出力(例えば、両面印刷されないジョブ、用紙上に斑点、縞、帯が出るなど)のすべてのカテゴリを取り扱う。いくつかのタイプの期待外出力に関しては、データフローにおける何らかのコンポーネントに起因する損なわれたデータが原因である場合があり、従って、データフロー・コンポーネントがこのコンポーネントの親である。

【0152】エラーコード・モデル24は、プリンタのコ



ントロール・パネルで見ることができるすべてのタイプのエラーコードを取り扱う。いくつかのエラーコードについては、損なわれたデータが原因であるので、データフロー・コンポーネントがこのコンポーネントの親である。

【0153】その他モデル23は、プリンタ・エンジンからの雑音、遅い印刷、双方向通信に関する問題などのような、上記3つのモデルによってカバーされないその他のプリンタ誤動作を取り扱う。

【0154】データフロー・モデル21、期待外出力モデル22、エラーコード・モデル24およびその他モデル23は、SACS O印刷診断システムの4つのエラー・コンポーネントとも呼ばれる。

【0155】設定モデル25は、アプリケーション、プリンタ・ドライバ、ネットワーク・ドライバおよびコントロール・パネルの設定のような、印刷システムにおけるすべての可能な設定を表す。設定が動作を決定する場合が多いので、このコンポーネントは4つのエラー・コンポーネントの親である。

【0156】図5における4つのエラー・コンポーネントのそれぞれは変数を持ち、変数は、そのコンポーネントにおける他のすべての変数の子孫（下位オブジェクト）であり、基本的には、それらすべての論理ORである。従って、もしもサブコンポーネントのうちの1つに問題があれば、そのコンポーネントに問題があることをルート変数が標示する。具体的に述べれば、図5における問題変数20は、その4つの親（すなわちデータフロー・モデル21、期待外出力モデル22、エラーコード・モデル24およびその他モデル23）の論理ORであり、従って、これは、印刷システムに問題があるか否かを一般的に表す。

【0157】図6は、データフロー・モデル21のコンポーネントを示す。それぞれの円は、対象となっているコンポーネントが、どのようにして損なわれた出力または失われた出力／出力停止を引き起こすことがあるかというベイジアン・ネットワーク・モデルを表現する。図6に示されているように、データフローは、顧客が使用している設定に従って、4つの異なるルートをたどることができる。例えば、データフローには、(1)パラレル・ケーブルによってローカルのPCに接続しているプリンタへ直接進む経路、(2)プリンタ・サーバによって管理されるプリンタへのネットワーク経由での経路、(3)パラレル・ケーブルによってプリンタ・サーバに接続しているプリンタへのネットワーク経由の経路、または、(4)（例えばヒューレット・パッカード・カンパニーから供給されるJetDirectソフトウェアを使用することによって）ネットワーク経由の直接的なプリンタへの経路、がありうる。

【0158】印刷システム設定は、データフロー・モデル21における1つの変数であり、これにより、データフ

ロー・モデル21においては適切な経路だけが使用されていることを確実にすることができる。データフローにおけるコンポーネントのそれぞれは、データがOKなのか、損なわれているのか、（どこか先の段階で既に失われて）もはや存在しないのか、などというデータの状態を、前のコンポーネントから入力として受け取る。コンポーネントはサブコンポーネントを有し、サブコンポーネントは、フローのその箇所で、データの損壊、損失／停止を引き起こす可能性を有する。

【0159】図7は、データフロー・モデル21に関するベイジアン・ネットワークを一層詳細に示している。データフロー・モデル21は、データフロー問題ノード500、データフロー・コンポーネント・ノード501を含み、さらにコンポーネント、データフロー・ノード、データフロー、グローバル・トラブルシューティング・ステップおよびコンポーネント・ネットワークに関するノードを含む。コンポーネント・ノードは、ファイル問題ノード502、アプリケーション問題ノード503、プリンタ・ドライバ問題ノード504、スプーラ問題ノード505、オペレーティング・システム・リダイレクト問題ノード506、ネットワーク・ドライバ/カード問題ノード507、サーバ問題ノード508、待ち行列問題ノード509、プリンタ・ネットワーク・ケーブル問題ノード510、プリンタ・ネットワーク・カード問題ノード511およびプリンタ問題ノード512を含む。データフロー・ノードは、フロー・ノード1、フロー・ノード2、フロー・ノード3、フロー・ノード4、フロー・ノード5、フロー・ノード6、フロー・ノード7、フロー・ノード8、フロー・ノード9、フロー・ノード10、フロー・ノード11および出力ノード531を含む。以下の記述では、フロー・ノード1は単にフロー1、フロー・ノード2は単にフロー2などのように呼称される。

【0160】ノード513、514、515、516、517、518および519は、グローバルなトラブルシューティング・ステップのノードを表す。例えば、ノード513は、“プリンタのローカル接続は動作しているか？”というトラブルシューティング・ステップを表している。ノード514は、“プリンタになにか反応が見られるか？”というトラブルシューティング・ステップを表している。ノード515は、“印刷ジョブはスプーラを通過しているか？”というトラブルシューティング・ステップを表している。ノード516は、“別のアプリケーションでジョブ印刷は動作するか？”というトラブルシューティング・ステップを表している。ノード517は、“パラレル接続されたプリンタへの印刷は動作するか？”というトラブルシューティング・ステップを表している。ノード518および519は、その他のグローバル・レベルのトラブルシューティング・ステップを表している。更に、ノードの追加によって他のトラブルシューティング・ステップを表すことができる。

【0161】図7は、モデル化された4つのシナリオのうちの1つを示している。ここでは、ユーザは、ネットワーク・プリンタ・サーバで制御されているネットワーク・プリンタに印刷を行っている。"データフロー問題"と名付けられた問題定義ノード500は、データフローに問題があるか否かを基本的に表し、YESとNOの2つの状態を持つ。ノード500は、データフローのすべてのコンポーネントを定義するノード501"データフロー・コンポーネント"に対する親である。ノード501は、ファイル、アプリケーション、プリンタ・ドライバ、スプーラ、プ  
リントなどの各コンポーネントに関する状態、およびデ \*

"データフロー・コンポーネント" 501の状態	"データフロー問題"500 の状態	
	y e s	n o
ファイル	x <sub>1</sub>	0
アプリケーション	x <sub>2</sub>	0
プリンタ・ドライバ	x <sub>3</sub>	0
スプーラ	x <sub>4</sub>	0
...	...	0
プリンタ	x <sub>11</sub>	0
N/A	0	1

【0164】表15のすべての"x"は、各コンポーネントについてそれが問題の原因である確率を表している。上述のように定義された知識獲得プロセスから、データフローにおけるコンポーネントについての確率分布が得られ、その後それが、上記の確率分布において直接使用される。

【0165】データフローにおける各コンポーネントに関して、コンポーネント標識ノード502、503、504、50  
※30

"ファイル問題" 502の状態	"データフロー・コンポーネント"501状態				
	ファイル	アプ リケ ー シ ョ ン	...	プリン タ	N/A
y e s	1	0	0	0	0
n o	0	1	1	1	1

【0167】コンポーネント標識ノードのすべては、データがフローにおける何らかの箇所xで失われたならば、それはフローの後の箇所y>xにおいても失われるであろう、という事実を表す一連のノードであるデータ  
フロー構造の親である。この例として、"フロー4"に関 ★

"スプーラ問題"状態	y e s		n o	
	o k	損失	o k	損失
"フロー3"状態	0	0	1	0
"フロー4"状態	0	1	0	1

【0169】フローの最後に、ユーザが何らかの出力を得たかどうかを表す出力ノード531がある。このノードは、フローにおける他のノードと同様に構築される。

【0170】データフロー・モデル21におけるトラブル

\*ータフローに問題がない場合の特別の状態N/Aを持つ。

【0162】図7において、弧は、因果関係の方向とは反対の、問題(データフローの問題)から、原因(コンポーネント)および副次的な原因へと向かっている。しかしながら、例えば持続などを表現する場合には、必要に応じて弧は反転される。確率の仕様が一層簡単になるので、ここではこのような弧の方向が使用される。"データフロー・コンポーネント"501についての条件つき確率分布が次の表15に示されている。

【0163】

【表15】

※5、506、507、508、509、510、511および512は、問題がそのコンポーネントに起因するかどうかを表す。コンポーネント標識ノード502～512は、YESとNOという2つの状態を持つ。例えば、コンポーネント標識ノード"ファイル問題"502は、次の表16に示されるような条件つき確率分布を持つ。

【0166】

【表16】

★する条件つき確率分布を考慮すると、次の表17の通りとなる。

【0168】

【表17】

シューティング・ステップには2つのレベルがある。トラブルシューティング・ステップの第1のレベルは、グローバルなトラブルシューティング・ステップであって、これは、どのコンポーネントが問題を引き起こして

いるかを判断するのに使用される。トラブルシューティング・ステップの第2のレベルは、ローカルのトラブルシューティング・ステップであって、これは、特定のコンポーネントについてそのコンポーネントの原因のうちのどれが正しいものであるかを判断するのに使用される。ローカル・トラブルシューティング・ステップは、コンポーネント・ネットワーク・ノードの一部である。

【0171】グローバル・トラブルシューティング・ステップ513、514、515、516、517、518および519は、データフローにおけるコンポーネント・ノード502～512の10 一部および（または）フロー・ノードの一部と関連す \*

"フロー5"状態		o k		損失	
"プリンタ問題"状態		y e s	n o	y e s	n o
513状態	y e s	0	1 - ε	0	0
	n o	1	ε	1	1

【0173】表18における確率εは、ユーザがプリンタをローカルに正しく接続して印刷することができず、従って、たとえプリンタが完全に機能してジョブが"フロー5"に達したとしても、プリンタが動作し始めるか否か 20 確実ではない確率である。確率εは、アクションの不確実度と呼ばれるもので、上記知識獲得の段ですでに記述されたものである。同様に、グローバルおよびローカルのトラブルシューティングをモデル化するために必要な知識獲得はすでに上述された。

【0174】図7における最後の欄は、コンポーネント・ネットワーク・ノード520、521、522、523、524、525、526、527、528、529および530を含む。コンポーネント・ネットワーク・ノード520～530のそれぞれは、対象 30 となっているコンポーネントの原因をモデル化するベジアン・ネットワークを表す。

【0175】図8は、コンポーネント・ネットワークの2つの例を示している。図7では、これらコンポーネント・ネットワークは、省略されて示されており、図8 ※

"プリンタNWケーブル原因"601	"プリンタNWケーブル・コンポーネント"528の状態	
	y e s	n o
欠陥	x <sub>1</sub>	0
正しく取り付けられていない	x <sub>2</sub>	0
仕様外	x <sub>3</sub>	0
プラグの間違い (プリンタまたは壁)	x <sub>4</sub>	0
N/A	0	1

【0178】ノード602、603、604および605のそれぞれは、プリンタ・ネットワーク・ケーブル故障の原因を表し、YESとNOという2つの状態を持つ。これらのノードは、上述のノード502と同様の条件つき確率分布を持つ。

【0179】例えば、ノード602は欠陥ケーブルを表す。ノード603は、正しく取り付けられていないケーブ 50

\*る。例えば、"ローカルにプリンタを接続することが問題を解決するか"513は、"フロー5"および"プリンタ問題"512と関連づけられる。ジョブが"フロー5"に達していないか、または問題がプリンタ自体に起因するものならば、プリンタは、ローカルに接続するようには動作しない。これは、次の表18に示されているようなノード513に関する条件つき確率分布を用いてモデル化される。

【0172】

【表18】

※に、拡大して表示している。特に、図8において、プリンタ・ネットワーク (NW) ケーブル・コンポーネント・ノード528およびプリンタNWカード・コンポーネント・ノード529が拡大されている (以下の記述においてネットワークをNWと略称する)。

【0176】標識ノード528"プリンタNWケーブル・コンポーネント"は、YESとNOという2つの状態を持ち、プリンタ・ネットワーク・ケーブルが問題の原因であるか否かを標示する。これは、図7におけるノード501と同様に、プリンタ・ネットワーク・ケーブルの原因に対する確率分布を表す"プリンタNWケーブル原因"601の親である。次の表19は、ノード601の条件つき確率テーブルを示している。表19におけるxは、領域専門家から獲得されるプリンタ・ネットワーク・ケーブル故障についての原因に対する確率分布を規定する。

【0177】

【表19】

ルを表す。ノード604は、仕様外のケーブルを表す。ノード605は、プリンタまたは壁の正しくないプラグにおけるケーブルに関する問題を表す。図8には副次的原因が存在しないが、多くの状況においては、原因の副次的原因が存在する。

【0180】ノード606、607および608はトラブルシューティング・ステップを表現する。例えば、ノード606

は、トラブルシューティング・ステップ"NWケーブルの再取り付けが問題を解決するか?"を表す。ノード607は、トラブルシューティング・ステップ"NWケーブルが仕様にあることを確実にすることにより問題を解決するか?"を表す。ノード608は、トラブルシューティング・ステップ"NWケーブルが正しいプラグに接続されることを確実にすることにより問題を解決するか?"を表す。

【0181】標識ノード529"プリンタNWカード・コンポーネント"もまたYES、NOの2つの状態を持ち、プリンタ・ネットワーク・カードが問題の原因であるか否かを標示する。これは、図7におけるノード501と同様に、プリンタ・ネットワーク・ケーブルの原因に対する確率分布を表す"プリンタNWカード原因"610の親である。

【0182】ノード611~620のそれぞれは、プリンタ・ネットワーク・ケーブルの原因を表し、YESとNOという状態を持つ。これらのノードは、上述のノード502と同様に条件つき確率分布を持つ。

【0183】例えば、ノード611は欠陥カードを表す。ノード612はネットワーク非互換性を表す。ノード613はサポートされていないネットワーク・カードを表す。ノード614は不正確なネットワーク・カード設定を表す。ノード615はネットワーク・カードが正しく取り付けられていないことを表す。ノード616は他のネットワーク問題を表す。ノード617はプリンタの電源のオフを表す。ノード618はネットワーク・カード上のファームウェアを更新する必要があることを表す。ノード619はネットワーク・カード上のファームウェアが損なわれていることを表す。ノード620はネットワーク・カード上のNVRAMが損なわれていることを表す。

【0184】ノード621~627はトラブルシューティング・ステップを表す。例えば、ノード621は、トラブルシューティング・ステップ"プリンタがオンにされることを確実にすることにより問題を解決するか?"を表す。ノード622は、トラブルシューティング・ステップ"NWカード設定が正しいことを確実にすることにより問題を解決するか?"を表す。ノード623は、トラブルシューティング・ステップ"NWカードの再取り付けが問題を解決するか?"を表す。ノード624は、トラブルシューティング・ステップ"NWカードを別のスロットへ移動することにより動作するか?"を表す。ノード625は、トラブルシューティング・ステップ"NWカード上のファームウェアを再ロードすることが問題を解決するか?"を表す。ノード626は、トラブルシューティング・ステップ"NWカード上のファームウェアを更新することが問題を解決するか?"を表す。ノード627は、トラブルシューティング・ステップ"NWカードをデフォルトに再設定して再構成する

ことが問題を解決するか?"を表す。ノード629は、質問"NWカードはヒューレット・パッカード・カンパニー製のものか?"を表す。ノード628は、質問"NWカードはサポートされているか?"を表す。

【0185】(図5に示されている)期待外出力モデル22は、ユーザが期待したものと異なる出力をプリンタから得る状況に関するベイジアン・ネットワーク・モデルから構成される。そのような状況は、両面印刷されないジョブ、用紙上の斑点、あまりに薄い印刷等々多数存在する。それらのそれぞれは、図8において示されているデータフローのコンポーネントの場合と同様にベイジアン・ネットワークを用いてモデル化されることができる。

【0186】(図5に示されている)エラーコード・モデル24は、ユーザがプリンタのフロント・パネル上でエラーコードを見る状況に関するベイジアン・ネットワーク・モデルから構成される。それらの問題もまたデータフローのコンポーネントの場合と同様にモデル化されることができる。すなわち、そのようなモデルは、例えば、問題定義ノード、原因に対する確率を定義するノード、各ノードに関して原因が存在するか否かを表す標識ノード、副次的原因に対する確率を定義するノード、各副次的原因に関するノード、各アクションに関するノードおよび各質問に関するノードを含む。

【0187】次に、アクションおよび質問のモデル化を更に詳細に記述する。アクションは、それぞれが解決の確率を有する多数の原因に関連づけられる。1つのアクションAが原因 $C_1, \dots, C_k$ を解決することができるとすれば、 $p(A=yes | C_1=yes), \dots, p(A=yes | C_k=yes)$ というk個の確率を規定しなければならない。原因 $C_1, \dots, C_k$ のいずれも存在しないならば、問題を解決するAの確率を必ずしも規定する必要はない。これは、 $C_1, \dots, C_k$ が、Aによって解決されるすべての原因を含むからである。

【0188】図8に見られるように、アクションは、ベイジアン・ネットワークにおいて、アクションが解決することができる原因の子ノードとして表されている。従って、ベイジアン・ネットワークにおけるアクションについてのノードは、アクションが問題を解決することができるか否かを表す。原因の存在があるからこそ、それに関連づけられたアクションが問題を解決することができるので、原因ノードからアクション・ノードへ、という因果関係の方向は正しい。

【0189】原因 $C_1, \dots, C_k$ を解決するアクションAについての条件付き確率テーブルが次の表20のように示される。

【0190】

【表20】

51

52

C <sub>1</sub> 状態		yes				n o			
C <sub>2</sub> 状態		yes		n o		yes		n o	
C <sub>3</sub> 状態		yes	n o	yes	n o	yes	n o	yes	n o
A 状態	yes	. 5	. 5	. 5	p <sub>1</sub>	. 5	p <sub>2</sub>	p <sub>3</sub>	0
	n o	. 5	. 5	. 5	1-p <sub>1</sub>	. 5	1-p <sub>2</sub>	1-p <sub>3</sub>	1

表20において、 $p_i$ は次の式(1)のように定義されて  
いる。 \* 【0191】

\* 【数2】

$$p_i = P(A=\text{yes} | C_i=\text{yes})(1-I_A)$$

式(1)

【0192】このように、複数の原因が存在する上述のような不可能な構成のすべてに関して、単一故障仮定を用いて、均一分布が規定される。ただ1つの原因だけが存在する構成について、該原因が与えられた場合のアクションが問題を解決する確率 $p_i$ が、不確実度が差し引かれて規定される。

【0193】不確実因子は、ユーザがアクションAを間違えて実行し、解決できる潜在性があるのに問題を解決しない確率を表す。この不確実度を表現しなければならないとき、アクションの不確実度を組み入れることが必要となる。間違えて実行することが容易なアクションがあり、これは、ユーザから無効なデータを受け取るという高いリスクを与える(アクションが役立たなかった)。アクションが高い不確実度を持つ場合には、ユーザから供給されるデータを高い不確実度で取り扱われなければならない。高い不確実度は、アクションが問題を解決したことをユーザが発見する確率を減じるので、アクションが問題を解決するという確率から $I_A$ が減算される。

【0194】図9は、問題領域における原因の徴候(シンプトン)をモデル化する質問の例である。ノード51は、フロント・パネルのエラーコードが"トレイ2上昇"を示している問題領域を表す。ノード51は、トレイ2上昇の原因に対する確率分布を表すノード52の親である。ノード52~56のそれぞれは原因を表す。例えば、ノード52はトレイ上昇を表す。ノード53は、トレイが間違えて使用されていることを表す。ノード54は、欠陥トレイを表す。ノード55は、送り/切り離しローラーの取り付けが正しくないことを表す。ノード56は他の問題を表す。

【0195】ノード53は、間違えて使用されているトレイの副次的原因に対する確率分布を表すノード57の親である。ノード58~61のそれぞれは副次的原因を表す。例えば、ノード58は仕様外の用紙を表す。ノード59は、トレイの用紙が多すぎることを表す。ノード60は、用紙サイズ調整機構が正しく取り付けられていないことを表す。ノード61は、トレイの取り付けが正しくないことを表す。

【0196】ノード62は、"異常な雑音があるか?"とい

うトラブルシューティング質問を表す。

【0197】質問は、最もありそうな原因および最もなさそうな原因についての信念を修正する潜在性を持つ。質問のタイプに従って質問についての情報を取得する2つの可能な手段がある。例えば図9における"異常な雑音が聞こえるか?"という質問が、本質的に1つまたは複数の原因の徴候に属するとすれば、または本質的に1つまたは複数の原因の結果とみなされることができるとすれば、確率は、 $P(Q|C)$ として領域専門家によって最良に導出される。一方、もしも質問が原因の徴候または結果と関連しているとは思われない一般のタイプの質問であれば、図8のノード628の場合のように、確率は $P(C|Q)$ として領域専門家によって最良に導出される。

【0198】しかしながら、確率がどのように導出されようとも、すべての確率の情報が正しく表現されて、2つのタイプの質問について同じ組み合わせルールが使用されるのを確実にするため、それらは、ベイジアン・ネットワークにおいて同じ方法でモデル化されなければならない。

【0199】図9は、問題領域において原因に対する徴候についての情報を提供する質問の例を示す。ノード"異常な雑音が聞こえるか?"は、プリンタからの異常な雑音についてユーザからの情報を要求する。その情報は、用紙のいれすぎ、用紙サイズ調整ブラケットの不適切な取り付け、トレイの不適切な取り付け、または欠陥トレイなどという"トレイ2上昇"原因のいくつかによって潜在的引き起こされている可能性がある。

【0200】このタイプの質問は、その質問によって要求される徴候を引き起こす可能性がある原因の子ノードとしてモデル化される。可能な原因のそれぞれについて、徴候が現れる確率が規定され、また、規定された原因のいずれもが存在しないとしても徴候が存在するという確率(すなわち漏洩確率)が規定される。図9における質問ノード62に関して、条件つき確率テーブルが次の表21に示されている。

【0201】

【表21】

"トレイあふれ"状態		Yes (Y)								No (N)							
"紙サイズ..."状態		Y				N				Y				N			
"トレイ取付不良"状態		Y		N		Y		N		Y		N		Y		N	
"欠陥トレイ"状態		Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N
異常雑音は聞こえるか？	Y	.	.	.	.	.	.	.	$x_1$	.	.	.	$x_2$	.	$x_3$	$x_4$	y
	N	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5

【0202】表21において、不可能な組合せについては均一な確率が与えられ、ただ1つの原因だけが存在する組合せ $(x_1, \dots, x_4)$ については、徴候の確率は $p(Q = \text{yes} | C = \text{yes})$ である。表21における $y_i$ は、原因がなにも存在しない場合の徴候の確率である。この確率は、通常、 $x$ に比較して小さい。

【0203】このタイプの質問のモデル化においては、原因が徴候に及ぼす影響は独立していると仮定して、確率の最大値を使用することができるようにする。依存した影響を持つ原因が複数存在する場合には、徴候を引き起こすそれらの組合わされた確率で最大確率を置き換えればよい。ページアン・ネットワークにおける一般的質問の実現は、得られた確率の方向を反転させる必要があるので、一層複雑である。

【0204】図8におけるプリンタ・ネットワーク・カ

ードのモデルは、質問ノード628"ネットワーク・カードはサポートされているものか？"を含む。この質問にYESが回答されれば、原因"サポートされていないネットワーク・カード"の確率はゼロとなり、NOが回答されれば、確率は上がる。この質問は、1つの原因の確率に影響を及ぼすにすぎないが、例えば"プリンタから異常な騒音が聞こえるか？"という質問のように、複数の原因の確率に影響を及ぼす質問を容易に想像することができる。

【0205】次の表22は、図8におけるノード610"プリンタnwカード問題の原因"の状態に関して取得された確率を示す。これは、プリンタ・ネットワーク・カードにおける原因の確率を規定する。

【0206】

【表22】

“プリンタNWカード原因” (610) 状態	“NWカードサポートされているか” (628) 状態	—	yes	n o		
	“プリンタNWカード・コンポーネント” (609) 状態	—	yes	n o	yes	n o
	欠陥	u <sub>1</sub>		0		0
	NW O/Sカード非互換	u <sub>2</sub>		0		0
	サポートされていないNWカード	u <sub>3</sub>		0		0
	NW設定が不正確	u <sub>4</sub>		0		0
	取り付け不良	u <sub>5</sub>		0		0
	その他のプリンタ問題	u <sub>6</sub>		0		0
	プリンタ電源オフ	u <sub>7</sub>		0		0
	NWカード上のファームウェア更新必要	u <sub>8</sub>		0		0
	NWカード上のファームウェアの損壊	u <sub>9</sub>		0		0
	NWカード上のNVRAM不良	u <sub>10</sub>		0		0
	N/A	0	0	1	0	1

【0207】表22において、uは、ネットワーク・カードについて何かしら具合の悪いところがあるということがわかった時の原因の確率を示しており、これは質問ノード628に対して無条件確率である。

【0208】x<sub>3</sub>およびy<sub>3</sub>は、それぞれYESおよびNOを回答される質問に条件づけられている第3の原因(サポートされていないネットワーク・カード)の確率である。質問が直接関連する原因についての確率の変化だけを領域専門家は規定するので、残りのフィールドはリンクのままとされている。このケースでは、ネットワー

\*ク・カードがサポートされているものであるということがわかれば、原因が“サポートされていないネットワーク・カード”である確率はゼロに近づき(不確実度のためゼロではない)、ネットワーク・カードがサポートされていないものであるということがわかれば、この確率は無条件確率u<sub>3</sub>より高くなる。

【0209】また、領域専門家は、次の表23に示されているような、質問の回答の確率P(Q)を与える。

【0210】

【表23】

“NWカードはサポートされているか” (628) 状態	
y e s	n o
0 . 9	0 . 1

【0211】Qに関連する原因セットがS<sub>Q</sub>(このケースでは、“サポートされていないネットワーク・カード”)

とラベルをつけられるとすれば、P(S<sub>Q</sub>)、P(S<sub>Q</sub>|Q)およびP(Q)についての情報が判明する。確率計算は次※

$$P(S_Q) = \sum_Q P(S_Q|Q)P(Q)$$

式(2)

【0213】領域専門家が互いに独立した3つの確率分布を規定したので、情報が矛盾する、すなわち上記式が、導出された確率にあてはまらないことがある。

【0214】これは、領域専門家に整合性の検証を実行する機会を与え、これらの確率を提供する際に何らかの

※の式(2)によって示される。

【0212】

【数3】

追加のガイダンスを与える。3つの導出された確率のうち最も正確なものは、おそらく原因の無条件確率P(S<sub>Q</sub>)であり、領域専門家は、式(1)があてはまるまでP(S<sub>Q</sub>|Q)およびP(Q)の両方またはいずれかを修正する。また、領域専門家が選択することができる2つ以

上の分布の正しい修正を自動的に提案することも可能である。

【0215】領域専門家による提案の選択または段階的改訂を通して得られる値  $P(S_0)$ 、 $P(S_0|Q)$  および  $P(Q)$  に整合性があると仮定すると、ベイズの公式による \*

$$P(Q|S_0) = P(Q|S_0, I_{C_0}) = \frac{P(S_0|Q, I_{C_0})P(Q)}{P(S_0|I_{C_0})} \quad \text{式(3)}$$

【0217】式(3)において、 $I_{C_0}$  は、原因が、yes および no という状態を持つ  $S_0$  セットにあるか否か (例えば図8のノード528および529参照) を規定する標識変数である。 $S_0$  における原因の確率分布は、この標識変数に従う。それらは、標識変数が yes であれば、領域専門家によって規定された確率を持ち、そうでなければ確率はゼロである。質問の事前確率  $P(Q)$  は、標識変数が yes であろうと no であろうと、同じである、すなわち  $P(Q) = P(Q|I_{C_0})$  であると仮定する。

【0218】 $S_0$  における原因のうちの1つが存在するとわかれば、 $I_{C_0} = \text{yes}$  であり、 $S_0$  における原因のいづれもが存在しないとわかれば、 $I_{C_0} = \text{no}$  である。従って、 $S_0$  が与えられたとすると、 $Q$  は  $I_{C_0}$  から独立している。これは、式(3)に示されているように、 $P(Q|S_0) = P(Q|S_0, I_{C_0})$  であることを意味する。

【0219】このように、関連する原因 ( $S_0$ ) の子であるノード  $Q$  を作成し、式(3)の確率を使用することによって、ベイジアン・ネットワークの形態において質問を実現することができる。

【0220】同じセットの原因に影響を及ぼす複数の質問がある場合、原因を所与とすればそれらは通常独立しているの、それらの確率を互いに独立して導出することが可能であり、式(3)によって規定されるようにそれらを互いに独立して実現することができる。

【0221】同じセットの原因に影響を及ぼす複数の相互に依存する質問がある場合、質問のすべてを接合した構成 (joint configuration) を所与として、原因の確率を規定することが必要である。従って、今や、従属的な質問セット  $Q$ 、および関連する原因セット  $S_0$  が存在する。領域専門家は、確率分布  $P(S_0)$ 、 $P(Q)$  および  $P(S_0|Q)$  を導出し、質問と原因の間の弧を、 $P(Q|S_0)$  を取得するために反転する。

【0222】セット  $Q$  における質問の数が少なければこの方法で十分であるが、いくつかあれば、共同の状態空間を含む単一ノードとしてではなく、ベイジアン・ネットワークにおける依存した構造で質問を表現する方がよい場合がある。このケースでは、質問と原因の間の弧の方向を逆転させるため、多数の一層複雑な弧の反転を実行しなければならない。

【0223】また、状況が一層複雑なケースでは質問は、いくつかの原因セットに影響を与えることがある。

\* 推論の方向を、次の式(3)に示されるように反転することができる。

【0216】  
【数4】

10 このような状況においては、例えば、質問  $Q$  と、関連する  $S_0^1$  および  $S_0^2$  という2つの原因セットがある。この場合、 $P(S_0^1 | I_{C_0}^1)$ 、 $P(S_0^2 | I_{C_0}^2)$ 、 $P(S_0^1 | Q, I_{C_0}^1)$ 、 $P(S_0^2 | Q, I_{C_0}^2)$  および  $P(Q)$  という5つの確率分布が存在する。

【0224】図10から図14は、質問から原因への方向を反転させる一連の弧反転を例示している。弧反転は、 $A$  の親を  $B$  に加えまたその逆を実行しながら、 $A$  から  $B$  へとベイジアン・ネットワークにおいて弧を反転させる操作である。R. D. Shachter 著 "Evaluating Influence Diagrams, Operations Research 34 (6), 871-882 (1986)" を参照されたい。

【0225】図10から図14において、原因  $S_0^1$  および  $S_0^2$  という2つのセットに影響を及ぼす質問  $Q$  がある。 $S_0^1$  は原因  $C_1$  および  $C_2$  を含み、 $S_0^2$  は原因  $C_3$  および  $C_4$  を含む。従って、 $Q$  は、 $C_1$ 、 $C_2$ 、 $C_3$ 、 $C_4$  のどのサブセットにも影響を及ぼす可能性がある。

【0226】図10は、弧71および72が質問から原因へ向かっている初期ベイジアン・ネットワークを示している。図11は、弧81の反転を示し、そのプロセスにおける  $S_0^1$  のすべての親 ( $I_{Q^1}$ ) の  $Q$  への追加、すなわち弧73の追加を示している。前述のように、 $Q$  は、 $S_0^1$  を条件として  $I_{Q^1}$  から独立しているの、図12において弧73を削除することができる。

【0227】図13は、 $Q$  から  $S_0^2$  への弧82の反転を示し、そのプロセスにおける  $S_0^2$  の親 ( $I_{Q^2}$ ) の追加 (弧84)、および  $Q$  の追加された親 ( $S_0^1$ ) の  $S_0^2$  への追加 (弧83) を示している。この場合も、 $Q$  は、 $S_0^2$  を条件として  $I_{Q^2}$  から独立しているの、弧84を削除することができる。同様に、 $I_{Q^2} = \text{yes}$  であれば  $S_0^1$  が no 状態に定められ、 $I_{Q^2} = \text{no}$  であれば  $S_0^2$  が状態 no に定められて、 $S_0^1$  が  $S_0^2$  に対する影響を持たないので、 $S_0^2$  は  $I_{Q^2}$  を条件として  $S_0^1$  から独立しており、よって弧83を削除することができる。この結果が、図14に示されている。

【0228】一般的に、複数のセットに影響を与える質問があれば、標識変数から質問への付加的な弧が追加され、および原因間の付加的な弧が追加される。すべてのこれらの弧は、ベイジアン・ネットワークにおける独立した関係に起因して削除されることができる。

【0229】図5に示されているように、ベイジアン・ネットワーク・モデル21から25の間にはいくつかの相互



関係がある。図5から理解されるように、データフローは、期待外出力、エラーコードおよびその他のモデル・カテゴリにおける"原因"となりうる。例えば、期待外出力のいくつかのカテゴリは、データフローのどこかで印刷ジョブが損なわれたことによって引き起こされる可能性がある。すなわち、期待外出力カテゴリに関するモデルにおける原因の1つとして、データフローの"損なわれた出力"の問題定義ノードがある(図7のノード500)。そこで、結合されたモデルは、直ちに単一トラブルシューティング・モデルとして働き、期待外出力カテゴリの 10 トラブルシューティング・ステップを、データフローのステップと結合する。

【0230】問題の共通の原因または原因の共通の副次\*

"C <sub>1</sub> 副次的原因"状态		S <sub>1</sub>			S <sub>2</sub>			N / A		
"C <sub>2</sub> 副次的原因"状态		S <sub>1</sub>	S <sub>2</sub>	N/A	S <sub>1</sub>	S <sub>2</sub>	N/A	S <sub>1</sub>	S <sub>2</sub>	N/A
"S <sub>2</sub> "状态	yes	0	0	0	0	1	1	0	1	0
	no	1	1	1	1	0	0	1	0	1

【0233】このように、両方の親がS<sub>2</sub>と一致する時、または一方がS<sub>2</sub>で他方が無である時、ノードS<sub>2</sub>は 20 状態yesとなる。このモデルにおいて、C<sub>1</sub>を所与としてS<sub>2</sub>が基礎をなす原因である確率、およびC<sub>2</sub>を所与としてS<sub>2</sub>が基礎をなす原因である確率が規定されている。従って、S<sub>2</sub>が存在する真の確率は、S<sub>2</sub>がC<sub>1</sub>を引き起こす確率とS<sub>2</sub>がC<sub>2</sub>を引き起こす確率の和である。

【0234】プリンタ領域におけるこの形態の例は、データフロー・モデルにおける何カ所かで(例えばアプリケーション、プリンタ・ドライバおよびネットワーク・ドライバにおいて)印刷ジョブの損壊を引き起こす潜在性を持つ"オペレーティング・システムの損壊"である。 30 しかしながら、単一故障仮定が与えられているとすると、オペレーティング・システムの損壊は、これらのうちの1箇所でのみ印刷ジョブの停止を引き起こすことができるにすぎない。よって、この原因は、基本的には、※

"C の副次的原因"状態	"C" 状態	
	y e s	n o
S <sub>1</sub>	x <sub>1</sub>	0
S <sub>2</sub>	x <sub>2</sub>	0
S <sub>3</sub>	x <sub>3</sub>	0
N/A	0	1

【0237】副次的原因S<sub>1</sub>およびS<sub>2</sub>が偶然同時に存在することが既知であれば、表25にフィールドを追加し、次の表26に示される構成を容易に作成することが

\*的原因がある時、モデル間に他の相互関係が発生する可能性がある。たとえ単一故障仮定を設けているとしても、このような状況が存在する可能性はある。しかしながら、共通の原因は、一度に1つの問題を引き起こすことができるにすぎない。

【0231】この状況は図15に示されている。図15において、副次的原因S<sub>2</sub>は、原因C<sub>1</sub>およびC<sub>2</sub>両者の副次的原因であるが、単一故障仮定によって、この2つの原因のうち一度にいずれか1つだけを引き起こすことができるにすぎない。次の表24に、図15のS<sub>2</sub>についての条件付き確率テーブルが示されている。

【0232】

【表24】

※ジョブを停止させる可能性のある箇所それぞれのそれぞれについて1つの状態を持ち、それらの状態のうちの1つであるにすぎない。

【0235】特定のシナリオに関する上記手法において単一故障仮定を撤廃することは簡単である。上述の通り、ノードの各状態ごとに1つの原因が存在するように、1つの原因(または1つの問題の複数の原因)の副次的原因を表すノードが存在する。このノードが原因の同時発生に関する状態をも持つようこの手法を拡張することは非常に簡単である。例えば、原因Cが副次的原因S<sub>1</sub>、S<sub>2</sub>およびS<sub>3</sub>を持つとすれば、ノードは、原因Cを所与とする副次的原因の確率を表す必要がある。この例は次の表25のように示される。

【0236】

【表25】

できる。

【0238】

【表26】

"C の副次的原因" 状態	"C" 状態	
	y e s	n o
S <sub>1</sub>	x <sub>1</sub>	0
S <sub>2</sub>	x <sub>2</sub>	0
S <sub>3</sub>	x <sub>3</sub>	0
S <sub>1</sub> およびS <sub>2</sub>	x <sub>4</sub>	0
N / A	0	1

【0239】一般に、単一故障仮定が撤廃された場合には、原因のすべての可能な組合せについての確率が必要となり、その数は、原因の2倍となり、非常に膨大な数となりうる。領域専門家によるこのようなすべての組合せ確率の導出を避けるため、すべての原因が独立していると仮定し、原因の組合せの確率は、それぞれに生じる原因の確率を乗算することにより見いだすことができる。

【0240】図16は、ユーザによって経験されるトラブルシューティング・プロセスの概要を示している。ステップ700において、ワールドワイド・ウェブのユーザ・インタフェースを介して、または、ユーザ自身のコンピュータ上で実行可能なスタンドアローン型プログラムとして、トラブルシューティング機構がユーザによって起動される。

【0241】ステップ701において、ユーザ・インタフェースは、ユーザが経験している問題の識別へとユーザを誘導する。これは、期待外の出力などの識別のためのメニュー・システムまたは画像のようないくつかの方法で行われることができる。ステップ702において、ユーザは、トラブルシューティング機構のメイン・ループに入る。そこでは、トラブルシューティング機構がまずアクションまたは質問を提案し、ユーザはそれに応答する(これは、何らかの一連のアクションを実行した後でもよい)。ステップ703およびステップ704において、ユーザは、提案を実行することができない場合、または単にその時点での実行を望まない場合、提案を実行しないことを選択することができる。

【0242】ユーザがアクションを実行した後、ステップ709において、ユーザは、アクションが問題を解決したか否かをトラブルシューティング機構に入力する。アクションが問題を解決したとすれば、トラブルシューティング機構は、ステップ706において成功裏に終了するが、アクションが問題を解決しなかったならば、トラブルシューティング機構は、ステップ707において、提案されなかったアクションまたは質問が残っているか否か検査する。残っていないとすれば、ステップ708において、トラブルシューティング・プロセスは失敗裏に終了し、問題の解決策を見出すことはできない。

【0243】アクションまたは質問が残っているとすれば、ステップ702において、トラブルシューティング・プロセスは続行する。同じように、トラブルシューティ

ング機構が質問を提案したとすれば、ステップ704において、ユーザは、質問を実行しないことを決定することができる。さもなければ、ステップ705において、ユーザは質問に回答し、プロセスは続行する。ステップ707において、提案されなかったアクションまたは質問が残っているか否か検査される。

【0244】図17は、実行すべき最善の次のステップを選択するプロセスの概要を示している。図17は、基本的に、図16のステップ702およびステップ707を実現するために実行されるサブステップをカバーしている。

【0245】ステップ800において、まだ提案されていないアクションまたは質問が残っているか否か検査される。残っていないければ、ステップ805において、以前に先送りされたアクションまたは質問が残っているか否か検査される。残っていれば、ステップ806において、それらの中から最善のものが選択され、ステップ807において返される。以前に先送りされたアクションも質問もなければ、すべてが試されて提案すべきものが何もないので、ステップ808において何も返さない。

【0246】ステップ800において、提案されなかったアクションまたは質問がまだ残っていることがわかれば、ステップ801において、アクションおよび質問の最適シーケンスが計算される。ステップ802において、原因が識別されるまで、この最適シーケンスを実行するための期待コスト(すなわちECR)が計算される。ステップ803において、質問のそれぞれに関して、まず質問を行って、次にその質問に対する回答が与えられた場合におけるアクションの最適シーケンスを実行する期待コスト(すなわちECO)が計算される。ステップ804において、最適シーケンスのアクションを実行する前に尋ねることが最適である質問が存在するか否かを調べるため、質問のそれぞれについてのECOと最適シーケンスのECRの比較が実行される。ステップ807において、最終的に最適アクションまたは質問が返される。

【0247】上記引用Heckerman文献は、観察および修理されることのできるコンポーネントの概念を使用している。解決アクションという用語が、問題を解決することができるアクションのために使用され、情報収集アクションという用語が、トラブルシューティング・プロセスに情報を提供するアクションのために使用されている。本発明の解決アクションは、基本的に、Heckerman文献のコンポーネントの修理に対応しているが、相違し

ている点は、Heckerman文献では、修理をコンポーネントに直接関連づけているので、アクションおよび原因の間に1対1の対応関係が存在している点である。これは非現実的である。なぜならば、現実のアプリケーションの場合には、複数のアクションによって解決されることができ、原因と共に、複数の原因を解決することができるアクションが存在するからである。

【0248】例えば、図8において、原因615"プリンタ・ネットワーク・カードが正しく取り付けられていない"は、アクション623"ネットワーク・カードの再取り付けが問題を解決するか？"およびアクション624"ネットワーク・カードをプリンタの中の別のスロットへ移動することにより動作するか？"という2つのアクションによって解決され得る。また、ネットワーク・カードを再び取り付けるというアクション623は、原因615"プリンタ・ネットワーク・カードが正しく装着されていない"および原因617"プリンタがオフになっている"という2つの原因を解決することができる。

【0249】原因とアクションの多対多の関係がモデルに組み込まれた場合、一層複雑なシナリオをモデル化することが可能となる。しかしながら、そのためには、Heckerman文献のアルゴリズムには修正が必要とされる。

【0250】Heckerman文献のアルゴリズムにおいて \*

$$\frac{P_1}{C_1} > \frac{P_2}{C_2} > \frac{P_3}{C_3} > \frac{P_4}{C_4} > \frac{P_5}{C_5} > \dots > \frac{P_N}{C_N}$$

式(4)

ただし  $p_i = p(A_i = \text{yes})$  である。

【0255】このように、最適アクションを、このシーケンスにおける最初のアクションとして見つけることができる。しかしながら、これがうまく作用するのは、複数アクションが独立している時だけであり、そのような場合を図18に示すが、これは本発明が適用されるケースではない。

【0256】前述した通り、上記シーケンスのアクションを実行する時の修理期待コスト(ECR)を、最初に観察を実行して次にその結果に基づいて最適シーケンスの\*

$$ECR = C_1 + (1-p_1)C_2 + (1-p_1-p_2)C_3 + \dots + (1-p_1 - \dots - p_{N-1})C_N \quad \text{式(5)}$$

【0259】Heckerman文献は、アクションとコンポーネントの1対1の対応関係を持ち、また単一故障仮定に従うので、 $p_1, \dots, p_N$ の和は1となり、 $p_i = p(A_i = y, A_j = n, j \neq i)$ である。複数の原因を解決することができるアクションが存在する時、問題を解決するアクションの和が1を超えるので、式(5)を使用することはできない。また、1より多くの原因を解決することができるアクションがある時、式(5)における単純な公式は十分でない。

【0260】図18に示されているように、3つのアクション( $A_1, A_2, A_3$ )によって解決されることができ、4つの原因( $C_1, C_2, C_3$ および $C_4$ )がある状況を考

\*は、 $p_i$ がコンポーネントが壊れている確率であり、 $C_i$ がコンポーネント*i*が壊れているかを調べるコストであるとすれば、次に考慮すべき最適なコンポーネントは、最も高い比率 $p_i/C_i$ を持つコンポーネントである。

【0251】原因(コンポーネント) $C_i$ を確率 $p(A_i = \text{yes})$ で解決するアクション $A_i$ があるとする、 $C_i$ が $A_i$ を実行するコストであるとした場合、比率は、 $p(A_i = \text{yes})/C_i$ と表すこともできる。アクション $A_i$ を実行することは、コンポーネント $C_i$ が壊れている(すなわち原因 $C_i$ が存在する)か否かを(一定の不確実度をもって)観察することを意味する。

【0252】一般的には、複数の原因を解決することができる1つのアクション $A_i$ について、故障しているコンポーネント(原因)が修理され問題が除かれる確率を $p(A_i = \text{yes})$ が表している、この場合も比率を $p(A_i = \text{yes})/C_i$ と表現することができる。

【0253】このように、独立した解決アクションからなる最適シーケンスを、次の式(4)に示されるようにアクションを分類することによって見つけることができる。

【0254】

【数5】

※アクションを実行する期待コスト(ECO)と比較することによって、最適の質問を見出すことができる。質問に関するECOがECRより低ければ、最初に観察を行うことが最善であるのは明らかである。

【0257】Heckerman文献は、次の式(5)に示されているような最適シーケンスの修理の期待コスト(ECR)を既述している。

【0258】

【数6】

察して見る。アクションは、関連する原因を解決する確率1を持つと仮定される。すなわち、 $p(A_i = \text{yes} | C_1 = \text{yes}) = 1$ 、 $p(A_i = \text{yes} | C_2 = \text{yes}) = 1$ 、 $p(A_2 = \text{yes} | C_2 = \text{yes}) = 1$ 等々である。次に、アクションが問題を解決する確率は、それが解決する原因の確率の和である。図18には、原因およびアクションの確率が示されている。アクションが等しいコストを持っていると仮定すれば、式(4)に従った最適シーケンスは、 $A_2, A_3, A_1$ である。しかしながら、これは、 $A_2$ の実行が失敗した時、正しくない。その場合、原因 $C_2$ および $C_3$ の確率はゼロとなり、 $A_1$ および $A_3$ の確率はそれぞれ0.2および0.15に変わり、従って最適シーケンスが $A_2, A_1, A_3$ となるか

らである。一般的には、前のアクションの実行が失敗した時にアクションの確率が変わることを考慮に入れることが必要である。これは、上記式(5)を次の式(6)のよ\*

\*うに書き替えることによって得られる。

【0261】

【数7】

$$\begin{aligned} ECR = & C_1 + (1 - p(A_1=y))C_2 + \\ & (1 - p(A_1=y) - p(A_2=y, A_1=n))C_3 + \\ & (1 - p(A_1=y) - p(A_2=y, A_1=n) - p(A_3=y, A_1=n, A_2=n))C_4 + \dots + \\ & (1 - p(A_1=y) - \dots - p(A_{N-1}=y, A_{N-2}=n, \dots, A_1=n))C_N \end{aligned} \quad \text{式(6)}$$

式(6)は次のように式(7)へ変換することができる。

※【数8】

【0262】

※10

$$\begin{aligned} ECR = & C_1 + (1 - p(A_1=y))C_2 + \\ & (1 - p(A_1=y) - p(A_2=y | A_1=n) \times p(A_1=n))C_3 + \\ & (1 - p(A_1=y) - p(A_2=y | A_1=n) \times p(A_1=n) - p(A_3=y | A_1=n, A_2=n) \times p(A_2=n | A_1=n) \times \\ & p(A_1=n))C_4 \\ & + \dots + \\ & (1 - p(A_1=y) - \dots - p(A_{N-1}=y | A_{N-2}=n, \dots, A_1=n) \times p(A_{N-2}=n | A_{N-3}=n, \dots, A_1=n) \times \\ & \dots \times \\ & p(A_1=n))C_N \end{aligned} \quad \text{式(7)}$$

【0263】このように、式(7)を計算する時、各アクションについて初期的には確率 $p(A_i=\text{yes})$ があるにすぎないので、アクションについての確率を段階を追って更新することが必要である。具体的にのべれば、 $p(A_1=\text{yes})$ を見つけ出し、 $A_1=\text{no}$ という証拠を挿入してすべての信念を更新し、 $p(A_2=\text{yes} | A_1=\text{no})$ を見つけだし、 $A_2=\text{no}$ という証拠を挿入してすべての信念を更新し、 $p(A_3=\text{yes} | A_2=\text{no}, A_1=\text{no})$ を見つけ出し、... というように進めることによって、式(7)に従ってECRを計算するのに必要なすべての確率が得られる。

【0264】しかしながら、これは実際的でない。なぜならば、図16のアルゴリズムのサイクルのそれぞれにおいて、各質問の各状態について一度ずつECRが計算されなければならないからである。上記の式の計算の速度を上げるために使用することができる近似法がある。式(7)を計算する各ステップにおいて、最後のアクションが役立たなかった場合には、残りのアクションが問題を解決する確率を更新することが、基本的に必要とされる。

【0265】原因とアクションの間には単純な対応関係があり、したがって、各アクションについて、それぞれが解決の確率を持つ原因のリストがある。

【0266】例えば、図19には、3つの原因( $C_1, C_2$  および  $C_3$ )を解決することができる1つのアクション ★

$$\delta_{C,A} = \frac{p(C=y | A=n)}{p(C=y)}$$

【0270】次に、式(7)において、アクションの失敗があった場合の原因の確率は、式(8)の因子を原因の確率に乗算することによって、次の式(9)のように近似

★( $A_1$ )が示されている。確率 $p(C_1=y)$ 、 $p(C_2=y)$ 、 $p(C_3=y)$ および $p(A_1=y)$ が、対応するノードの横に示されている。こうして、

$$\begin{aligned} p(A_1=y) = & p(A_1=y | C_1=y \text{ or } C_2=y \text{ or } C_3=y) \times p(C_1=y \\ & \text{ or } C_2=y \text{ or } C_3=y) = p(A_1=y | C_1=y) \times p(C_1=y) + p(A_1=y | C_2=y) \times p(C_2=y) + p(A_1=y | C_3=y) \times p(C_3=y) \end{aligned}$$

が成立する。  
【0267】図19の例では、 $p(A_1=y | C_1=y) = .5$ 、 $p(A_1=y | C_2=y) = 1.0$ 、 $p(A_1=y | C_3=y) = 0.25$ であるので、 $p(A_1=y) = .5 \times 0.4 + 1.0 \times 0.2 + 0.25 \times 0.4 = .5$ である。

【0268】原因とアクションの間のこのような単純な関係を使用して、アクションが失敗した時( $A_i=n$ )の、原因に及ぼす影響を近似することができる。アクションが失敗した時、それが解決することができたであろう原因の確率は減少し、それに応じて、残りの原因の確率は、(単一故障仮定に従って)和が1を維持するように増加する。このように、トラブルシューティングを開始する前に、アクション(A)およびアクション(A)が解決することができる原因(C)のそれぞれについて、Aが失敗したときの $p(C=y)$ の相対的減少の起こりそうな比率が、次の式(8)のように記録される。

【0269】

【数9】

式(8)

される。

【0271】

【数10】

$$p(C=y|A=n,e) = \frac{p(C=y|e)p(A=n|C=y,e)}{p(A=n|e)} = \frac{p(C=y|e)p(C=y|A=n,e)}{p(C=y|e)} \approx \text{式(9)}$$

$$p(C=y|e) \frac{p(C=y|A=n)}{p(C=y)} = p(C=y|e) \times \delta_{C,A}$$

【0272】こうして、 $A_i$ が原因 $C_1, \dots, C_k$ を解決す \* 【0273】  
ることができる とすれば、以下の式が成立する。 \* 【数11】

$$\begin{aligned} p(A_i=y|A_j=n) &= p(A_i=y|C_1=y, A_j=n) \times p(C_1=y|A_j=n) + \dots + \\ & p(A_i=y|C_k=y, A_j=n) \times p(C_k=y|A_j=n) \\ &\approx p(A_i=y|C_1=y, A_j=n) \times p(C_1=y) \times \delta_{C_1, A_j} + \dots + \\ & p(A_i=y|C_k=y, A_j=n) \times p(C_k=y) \times \delta_{C_k, A_j} \end{aligned}$$

【0274】しかしながら、式(9)において、 $A_j=n$ と  
いう証拠が挿入されないままそれに応じて信念が更新さ  
れたとすれば、ベイジアン・ネットワーク・モデルから  
直接読み取ることができない確率  $p(A_i=y|C_1=y, A_j=$  \*

$n), \dots, p(A_i=y|C_k=y, A_j=n)$ が必要とされる。このよ  
うな確率もまた次の式(10)のように近似される。

$$p(A_i=y|C_1=y, A_j=n) \approx p(A_i=y|C_1=y) \quad \text{式(10)}$$

【0276】これは満足できる近似法である。なぜなら  
ば、原因が存在する( $C_1=y$ )という事実が、すべての他  
の原因の確率を(単一故障仮定に従って)強制的にゼロに  
し、それに続いてすべての原因の確率が決定され、すべ  
てのアクションの確率も決定されるからである。アクシ  
ョンのうちの1つが失敗したかどうかかわかれば、相違  
は生じない。

【0277】このように、式(9)および式(10)におけ  
る近似を使用して、いくつかのアクションによって解決  
されることができる複数の原因、およびいくつかの原因  
を解決することができる複数のアクションが存在する時  
の、アクションの最適シーケンスを実行するコストを、  
単一伝播の式(7)を使用して計算することができる。

【0278】ECRを計算するためには、最初にアクシ  
ョンの最適シーケンスを見つけることが必要である。E  
CRを求めるのに式(7)が使用される。また、次の表2  
7に示されるような段階を追ったアルゴリズムを使用し  
て望ましいアクション・シーケンスを見出すのに、式  
(7)を使用することができる。

【0279】

【表27】最も高い $p_i/C_i$ を持つアクションとして、★

$$ECO_Q = C_Q + ECR_Q = \text{yes} \times p(Q=\text{yes}|e) + ECR_Q = \text{no} \times p(Q=\text{no}|e) \quad \text{式(11)}$$

【0283】トラブルシューティング・モデルにおいて  
すべての質問に関する $ECO_Q$ を計算することは、各質  
問の各状態について1つの信念伝播を必要とする。これ  
は、例えば図7に例示されるデータフロー・モデルに関  
して、質問が20から30ある場合のように多くの質問  
がある場合には実際的ではなく、この計算は、Pent  
ium (登録商標) 166MHzのPC上で15秒のオー  
ダーを要し、これは、平均的なユーザが待つことができ  
る時間を超える。一層速いコンピュータを用いればこの

★アクション $A_i$ を最初に見出す。 $i=0$  からアクション  
数に対応する数(#actions)まで、以下を繰り返す

・  $A_{i-1}=\text{no}$ (失敗)を仮定して原因の確率を再計算する  
(式9を参照)

・ 最も高い $p(A_i=\text{yes}|A_1=\text{no}, \dots, A_{i-1}=\text{no})/C_i$ を持  
つアクションとして、アクション $A_i$ を見出す

【0280】しかしながら、複数のアクションが相互に  
依存しているという事実のため、このアルゴリズムは、  
最適シーケンスを見出すことを保証することはできな  
い。しかし、実際問題として、通常の場合このシーケ  
ンスは適切である。必要であれば、例えば動的プログラミ  
ングなどで実現されるnステップ先取り手法によって、  
このアルゴリズムを拡張することができる。

【0281】Heckerman文献のアルゴリズムにおいて  
は、まず質問Qを実行し、次にその質問に対する回答に  
基づく最適シーケンスのアクションを実行する期待コス  
ト( $ECO$ )に基づいて、質問の選択が行われる。たとえ  
ば、期待コストは、コスト $C_Q$ と回答yes、noを有する  
質問Qについて、式(11)のように定義される。

【0282】

【数13】

問題を解決する望みはあるが、1つのモデルに一層多く  
の質問が含まれるとすれば、問題は指数関数的に増幅す  
る。

【0284】状態sと証拠eの質問Qの観察が与えられ  
た場合、問題を解決するアクションの確率 $p(A=\text{yes}|$   
 $Q=s, e)$ を事前に計算しなければならないことにより、  
問題が生じる。この確率は、次の式(12)に示されるよ  
うに、ベイズの公式および単一故障仮定を利用すること  
によって近似することができる。

【0285】

【数14】

$$p(A = \text{yes} | Q = s, e) = \frac{p(Q = s | A = \text{yes}, e) \times p(A = \text{yes} | e)}{p(Q = s | e)} \approx \frac{p(Q = s | A = \text{yes}) \times p(A = \text{yes} | e)}{p(Q = s | e)} \quad \text{式(12)}$$

【0286】この近似は、 $p(Q=s|A=\text{yes},e)$ を $p(Q=s|A=\text{yes})$ で近似し、それによって前の証拠 $e$ からのいかなる影響をも無視することにより、原因およびアクションが密接に関係づけられるという事実を利用する。これは良好な近似である。なぜならば、特定のアクション $A$ が問題を解決することが既知であれば、高い確率で、 $A$ によって解決される原因のうちの1つが真の原因であることもまた既知であるからである。単一故障仮定によって、1つの原因について高い確率を持つことは、残りの原因に対しては非常に低い確率を直接意味するので、原因は、それらの"yes"または"no"のいずれかの状態に固定される。原因が与えられた時にアクションが質問から切り離されるので、他の質問または他のアクションに関するいかなる付加的情報も、 $A$ が"yes"状態で与えられた時の質問 $Q$ に対する影響をほとんど持たないことがわかる。

【0287】経験によれば、SACSOプロジェクトの間に構築されるモデルのように、1つのアクションに対して解決される原因の数は1.9であり、これは、アクションが問題を解決するのに既知であるときには、ほとんどすべての確率が1つまたは2つの原因に集中されることを意味する。

【0288】データフロー・モデルに関して言えば、式(12)における近似によって、すべての質問に関する $ECQ$ の計算が、データフロー・モデルにおいて1秒未満で実行されることが可能となる。

【0289】Heckerman文献アルゴリズムにおいては、\*30

$$p_{Qidc} = \max_C \max_s \frac{p(C = y | Q = s) - p(C = y)}{1 - p(C = y)} \times p(Q = s) \quad \text{式(13)}$$

【0294】式(13)は、原因 $C$ および $Q$ の状態 $s$ を識別する。この場合、状態 $s$ における $Q$ を観察する時に $C$ を識別する確率は、 $Q$ のすべての原因およびすべての状態を考慮する時に可能な最も高い値となる。

【0295】式(13)の第1項(分数部分)は、現在値から1への確率の可能な増加に対して相対的な $Q=s$ を観察する時の、原因の確率の増加を直感的に記述している。このように、相対的増加は0と1の間にある。

【0296】相対的増加は、質問 $Q$ への回答 $s$ が実際に得られる確率で乗算されなければならない。この確率が小さいと、状態 $s$ における質問 $Q$ を観察することにより原因 $C$ を識別する確率も、小さくなる。

【0297】式(13)に定義されているような、質問が原因を識別する確率を計算することによって、Heckerman文献における解決アクションの場合と同じ方法で質問(および情報収集アクション)を取り扱うことが可能である。前述したように、アクションが問題を解決する確率

\*アクションおよび質問の役割が異なる。アクションは問題を解決することができ、成功したならばトラブルシューティング・プロセスを終了させることができる。質問は、トラブルシューティング・プロセスのために情報を提供することができるだけであり、プロセスを終了させることはできない。このように、アクションが残されていなければ、質問は役に立たない。

【0290】質問(または情報収集アクション)は、問題を解決するアクションに対応する原因を時において識別することができるので、これは最適ではない。原因が識別された時、問題を解決するために何かなされるべきかはほとんど常に自明であり、自動トラブルシューティング機構によって提案されることのできる更なるアクションは(おそらく)なにもない。

【0291】Heckerman文献において質問(および情報収集アクション)が解決アクションと同等に取り扱われることができるという目標を達成するためには、アクションが問題を解決する確率と比較されることのできる原因を質問が識別する確率について、表現を見出すことが必要である。

【0292】質問 $Q$ (または情報収集アクション)が原因を識別する確率は、次の式(13)に示されるように、原因および質問の状態に対する最大値として表されることができる。

【0293】

【数15】

を $p_i$ とすれば、アクションは、それらの比率 $p_i/C_i$ で分類される。質問が問題を解決する確率が、質問が原因を識別する確率であるとみなされるならば(つまり、 $p_j = p_{Qidc}$ )、質問は、アクションと同様に直接取り扱われ、同じシーケンスに分類されることができる。また、アクションおよび質問の最適シーケンスの期待コストを計算する時、質問は、アクションと同様に取り扱われることができる。

【0298】この考え方によって多くが得られる。実際、Heckerman文献アルゴリズムにおいて、質問の取り扱いは近視的(myopic)である(すなわち、1ステップ先取りが行われるだけである)。この方法では、最適シーケンスのいずれの位置にも質問を置くことができるので、得られるのは限定された $n$ ステップ先取りである。これは、完全な $n$ ステップ先取りではない。なぜならば、質問が原因を識別する確率だけが考慮され、一層最適なトラブルシューティング・シーケンスを可能にする

ことがある価値ある情報を提供する質問の能力が考慮されないからである。

【0299】また、質問(または情報収集アクション)は残されているが解決アクションがないような状況においても何かが得られる。この状況において、Heckerman文献アルゴリズムは、たとえ質問の1つが原因を識別することができたとしても、停止する。アルゴリズムへのこのような追加は、原因に関心が集中するのに質問が役立つ限りは、質問することを続行する。

【0300】Heckerman文献アルゴリズムにおいては、質問が最適トラブルシューティング・シーケンスに十分な影響を与えるとすれば、それが提案され、質問が行われなかった場合より低いコストが得られるようにする。例えば、質問することにより、最適シーケンスにおいてアクション4と5が交換されるとすれば、質問は適切であり、シーケンスの一層安い期待コストが与えられる。しかし、このルールに基づくトラブルシューティング・プロセスは、後続の複数の手順に影響を持たない質問を行う可能性があるため、混乱を生む。

【0301】これを軽減する1つの方法は、提案される次のステップにおいて潜在的に相違を生むことがある場合にだけ、質問を行うことを許容することである。これは、簡単に実行することが可能であり、ユーザから見て一層直感的で整合性のあるトラブルシューティング・プロセスを与える。

【0302】アルゴリズムのこのような変更は、ユーザに一層論理的なシーケンスを提供するが、将来さらにアクションを再配置することができるような質問を行うことは許容されないため、最適性が減少する。このように、表面的にはより論理的なシーケンス・ステップになるとはいえ、平均すると、わずかに長く、一層高価なシーケンスが得られることとなる。

【0303】問題領域および知識獲得(オーサリング)のモデル化の基礎をなす仮定がいくつか存在する。第1は、単一故障仮定である。一般的に、完全なシステムにおいて常に単一の故障だけがあると仮定される。この仮定は、トラブルシューティング・アルゴリズムおよび知識獲得プロセスの妥当性にとって重要である。システムにおける故障の組合せについて複数の確率が評価されなければならないとすれば、領域専門家は、たとえ小さい問題領域の場合であっても数年間残業しなければならないであろう。しかし、上述のように、単純なケースでは単一故障仮定を撤廃することは可能である。そのような状況は、各ケースで明確に識別されなければならない。そのようなわずかな特定のケースにおいて単一故障仮定が取り除かれると、上述されたトラブルシューティング・アルゴリズムはもはや最適ではなくなるが、それでもなお、このアルゴリズムは、最も可能性のありそうな原因に向かってその原因を訂正し、さらに次の最も

可能性のありそうな原因に向かって進む。

【0304】問題領域および知識獲得(オーサリング)のモデル化の基礎をなす第2の仮定は、原因の独立性である。原因は、互いに独立して発生すると仮定される。これは、原因間の依存関係を考慮する必要がないので、知識獲得を非常に単純なプロセスにする。これはまた、原因間の因果関係(たとえば、原因が他の原因を引き起こしたりする)があり得ないことを意味する。これは、一般的に適切な仮定であるように見えるが、必ずしも常に有効ではない。例えば、プリンタにおけるプリンタ・トナー・カートリッジと他のコンポーネントの間の関係の場合、トナー・カートリッジが壊れればインクが他のコンポーネントに漏れるので、それらの間に因果関係が存在する。そのような関係をモデル化することは有益であるが、現在のフレームワークにおいてモデル化は容易ではない。

【0305】問題領域および知識獲得(オーサリング)のモデル化の基礎をなす第3の仮定は、「持続」である。一般的に、アクションの実行は、検討中のシステムの環境または構成を変えないし、観察は有効のままであると仮定される。この仮定が保たれない単純なケースを取り扱う方法が上述されている。

【0306】第4の仮定は、コスト・コンポーネントの加法性である。(時間、リスク、金額および侮辱)という4つのコスト・コンポーネントは加法的であると仮定される。現在までの経験によれば、コスト因子の比較的狭い間隔が使用される時、この仮定は少なくとも適切である。より大きい間隔が使用される場合には、線形性は多分保たれないであろう。例えば、時間コンポーネントは、100分が50分の倍以上に悪く指数関数的に増大し、金額コンポーネントは、1000ドルが500ドルの倍ほどには悪くならないようにやや指数関数的に増える。

【0307】第5の仮定は、「嘘」がないということである。アクションおよび質問について確率を取得する時、ユーザは、ユーザが正しいと思う回答を与えると仮定される。不正確な回答をユーザが故意に与えないと仮定される。この仮定は現実の世界では必ずしも当てはまらないが、その可能性を考慮した確率を与えることが非常に難しいので、この仮定は知識獲得にとって重要である。

【0308】本発明には、例として次のような実施様態が含まれる。

(1) システムの故障を引き起こすシステム・コンポーネント(209,210)をモデル化するペイジアン・ネットワーク(1-11,500-531,601-608,610-629)であって、上記システム・コンポーネントが故障を引き起こしているか否かを示す状態を持つ標識ノード(502-512)と、上記標識ノードに結合され、故障を生成するシステム・コンポーネントの原因をそれぞれが表す複数の原因ノード(602-605,611-620)と、上記複数の原因ノードのうちの少なく

とも1つの原因ノードに結合され、該結合された原因ノードのいずれかによって表される原因を修復するアクションを提案するトラブルシューティング・ステップをそれぞれが表す第1の複数のトラブルシューティング・ノード(606-608, 621-627)と、を備えるベイジアン・ネットワーク。

【0309】(2) 上記複数の原因ノードのうちの少なくとも1つの原因ノードに結合され、質問を表す質問ノード(628, 629)を備えるベイジアン・ネットワークであって、該質問が、回答するときに、該結合された原因ノードのいずれかによって表される原因に関する潜在的情報を提供する上記(1)に記載のベイジアン・ネットワーク。

【0310】(3) 上記複数の原因ノードが、上記システム・コンポーネントの故障の原因に対する確率分布を表す原因ノードを経由して標識ノードに結合される、上記(1)に記載のベイジアン・ネットワーク。

【0311】(4) 上記原因ノード(610)に結合され、上記システム・コンポーネントの故障の原因または徴候に必ずしも関連しない一般のタイプの質問を表す質問ノード(628)を備える、上記(3)に記載のベイジアン・ネットワーク。

【0312】(5) 第1のアクションを提案する第1のトラブルシューティング・ステップについて、ユーザが間違っ第1のアクションを実行する確率を表す不確実因子を使用して、上記第1のアクションが第1の原因を解決するか否かを計算する上記(1)に記載のベイジアン・ネットワーク。

【0313】(6) システムをトラブルシュートするために使用される知識獲得を実行する方法であって、

(a) トラブルシュートすべき問題を識別するステップと、(b) 上記問題の原因を識別するステップと、

(c) 上記原因の副次的原因を識別するステップと、

(d) トラブルシューティング・ステップを識別するステップと、(e) トラブルシューティング・ステップを原因および副次的原因に対応づけるステップと、(f) 上記ステップ(b)で識別された原因および上記ステップ(c)で識別された副次的原因について確率を評価するステップと、(g) 上記トラブルシューティング・ステップにおいて設定されたアクションおよび質問について確率を評価するステップと、(h) 上記トラブルシューティング・ステップにおいて設定されたアクションおよび質問についてコストを評価するステップと、を含む知識獲得を実行する方法。

【0314】(7) 上記トラブルシューティング・ステップのそれぞれが、(a) 問題を解決することができるアクションを提供する解決アクションと、(b) 情報収集アクションとのカテゴリのうちの1つにおけるアクションを含む上記(6)に記載の知識獲得を実行する方法。

【0315】(8) 上記ステップ(e)において、上記トラブルシューティング・ステップのそれぞれが、解決することができる原因または副次的原因に対応づけられたアクションと、関係する原因または副次的原因のいずれかに対応づけられた質問と、を含む上記(6)に記載の知識獲得を実行する方法。

【0316】(9) 上記ステップ(g)が、(g. 1) 関連する原因または関連する副次的原因のそれぞれについて、第1のアクションを実行することが問題を解決するか否かを示す第1の確率を求めるステップと、(g. 2) ユーザが上記第1のアクションを正しく実行する可能性を示す第2の確率を求めるステップと、(g. 3) ユーザに上記第1のアクションを実行するよう指示することが問題解決に結びつくか否かを定めるために上記第1の確率および第2の確率を組み合わせるステップと、を含む上記(6)に記載の知識獲得を実行する方法。

【0317】(10) 上記ステップ(h)において、上記第1のアクションについてのコストが、該第1のアクションを実行する時間と、該第1のアクションを実行する時に何かを壊すリスクと、該第1のアクションを実行するために必要ななんらかの部品を購入するために必要とされる金額と、および第1のアクションが提案される時にユーザが経験するかもしれない侮辱の程度と、を考慮に入れた因子を含む上記(6)に記載の知識獲得を実行する方法。

【0318】

【発明の効果】本発明によると、システムのトラブルシューティングに際して、問題を解決できる確率が最も高く、必要とされる期待コストの最も少ないアクションがユーザに提案される。

【図面の簡単な説明】

【図1】本発明の1つの好ましい実施形態に従ったトラブルシューティング環境のブロック図。

【図2】本発明の1つの好ましい実施形態に従ったウェブ・サーバのブロック図。

【図3】本発明の1つの好ましい実施形態に従ったトラブルシューティング・プロセスにおいて使用される顧客パーソナル・コンピュータ内のコンポーネントを示すブロック図。

【図4】本発明の1つの好ましい実施形態に従った知識獲得を実行するステップの流れ図。

【図5】本発明の1つの好ましい実施形態に従った自動顧客サポート運用印刷診断システム(SACSO)のブロック図。

【図6】本発明の1つの好ましい実施形態に従ったデータフロー・モデルのコンポーネントを示すブロック図。

【図7】本発明の好ましい実施形態に従った図6のデータフロー・モデルのためのベイジアン・ネットワークの詳細を示すブロック図。

【図8】本発明の好ましい実施形態に従ったコンポーネ



ント・ネットワークの2つの例を示すブロック図。

【図9】本発明の好ましい実施形態に従った、問題領域における原因の徴候をモデル化する質問の例を示すブロック図。

【図10】本発明の好ましい実施形態に従った、質問から原因へ方向を反転させる一連の弧反転を示すブロック図。

【図11】本発明の好ましい実施形態に従った、質問から原因へ方向を反転させる一連の弧反転を更に示すブロック図。

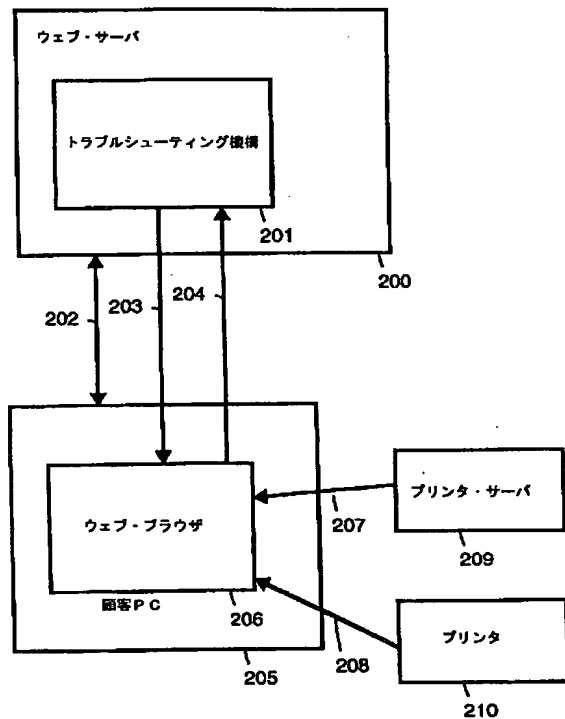
【図12】本発明の好ましい実施形態に従った、質問から原因へ方向を反転させる一連の弧反転を更に示すブロック図。

【図13】本発明の好ましい実施形態に従った、質問から原因へ方向を反転させる一連の弧反転を更に示すブロック図。

【図14】本発明の好ましい実施形態に従った、質問から原因へ方向を反転させる一連の弧反転を更に示すブロック図。

【図15】本発明の好ましい実施形態に従った、問題の共通の原因または原因の共通の副次的原因が存在する状

【図1】



況を示すブロック図。

【図16】ユーザによって経験される、本発明の好ましい実施形態に従ったトラブルシューティング・プロセスの流れ図である。

【図17】本発明の好ましい実施形態に従った、実行すべき最善の次のステップを選択するプロセスの流れ図。

【図18】3つのアクション(A<sub>1</sub>, A<sub>2</sub>, A<sub>3</sub>)によって解決されることが出来る4つの原因(C<sub>1</sub>, C<sub>2</sub>, C<sub>3</sub>, C<sub>4</sub>)が存在するケースを示すブロック図。

10 【図19】3つの原因(C<sub>1</sub>, C<sub>2</sub>, C<sub>3</sub>)を解決することができる1つのアクション(A<sub>1</sub>)が存在するケースを示すブロック図。

【符号の説明】

1、500、502、513、520、601、610  
ペイジアン・ネットワーク

209、210 システム・コンポーネント

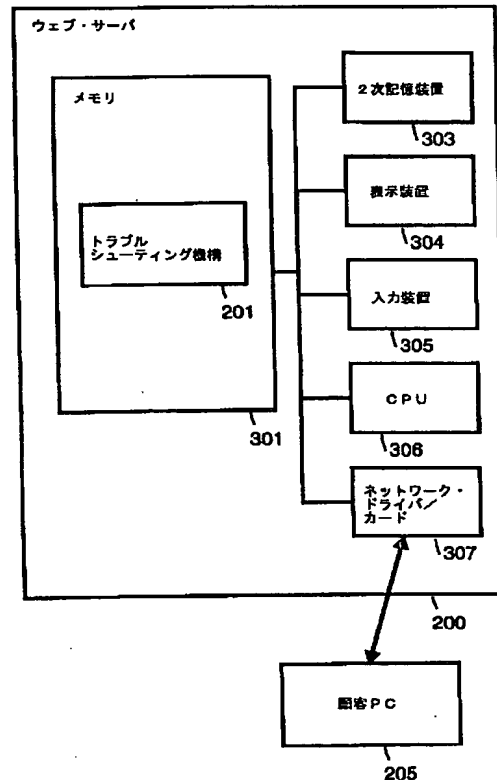
502、512 標識ノード

602、611 原因ノード

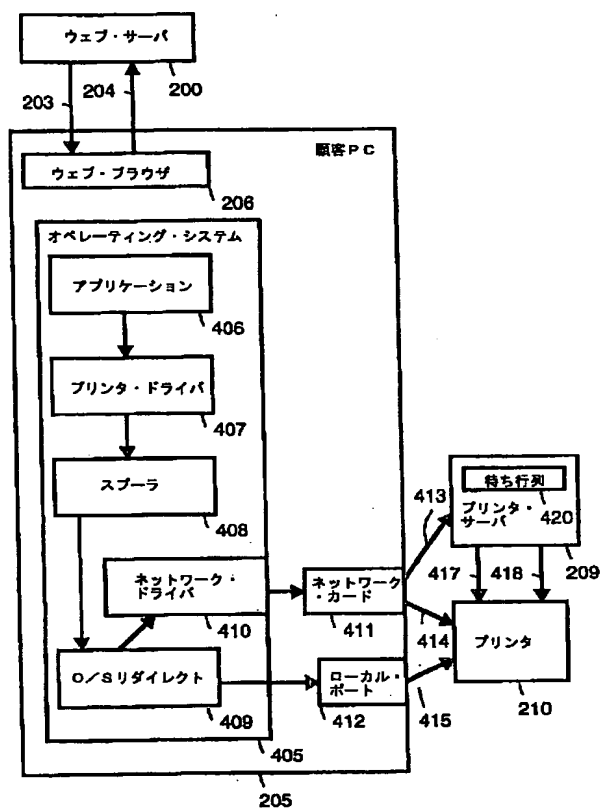
606、621 トラブルシューティング・ノード

20 628、629 質問ノード

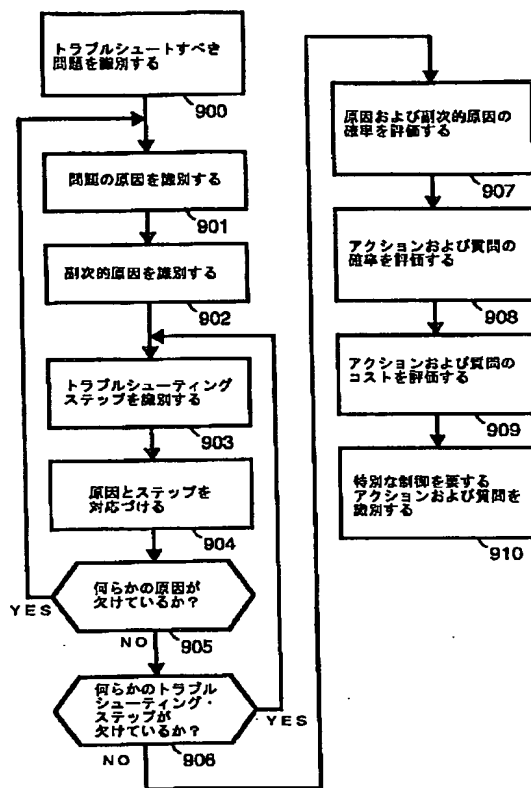
【図2】



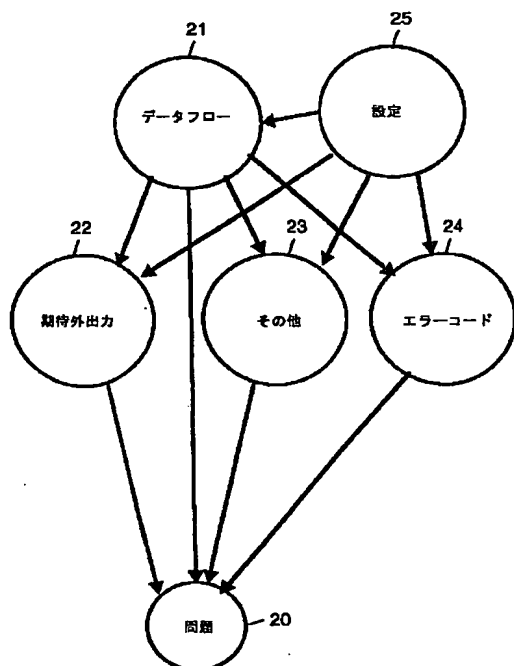
【図3】



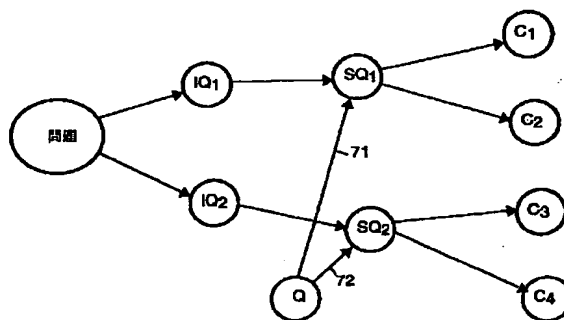
【図4】



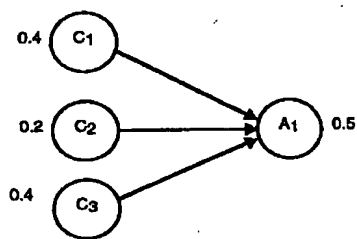
【図5】



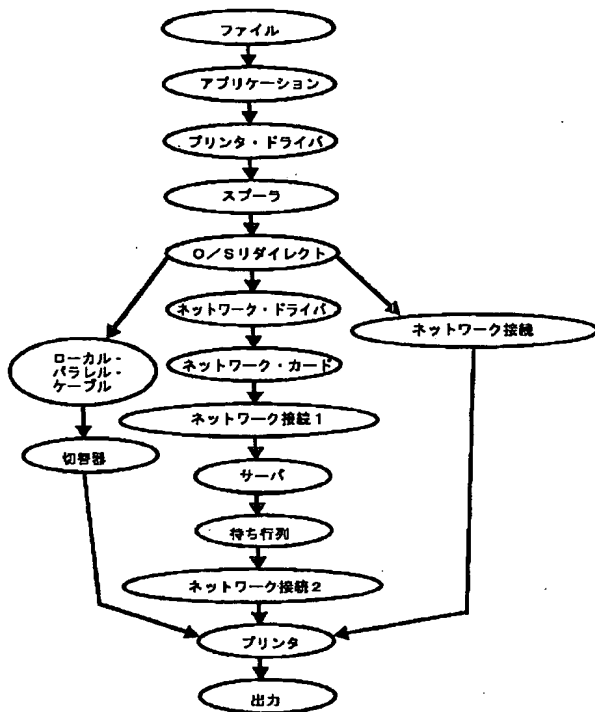
【図10】



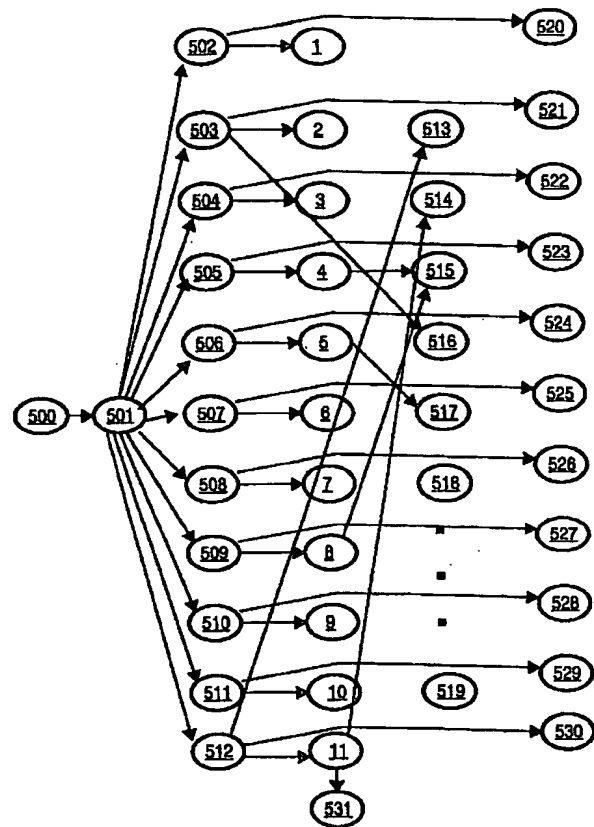
【図19】



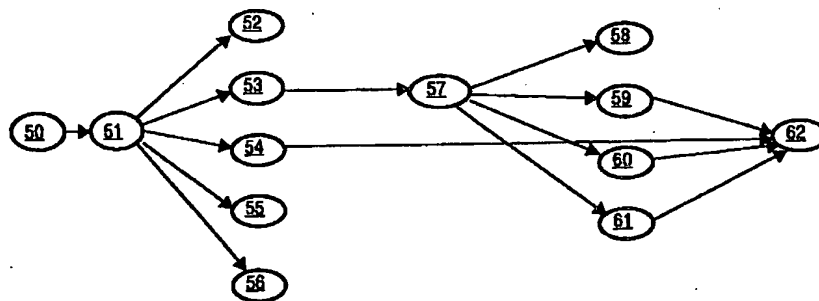
【図6】



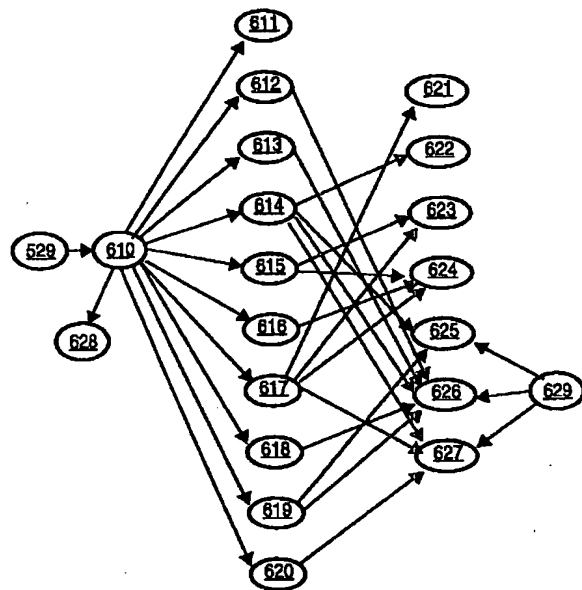
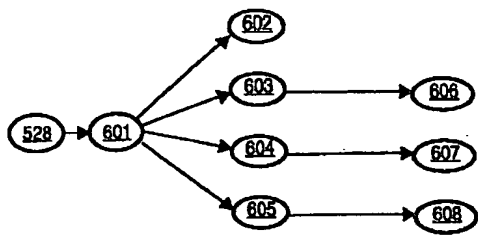
【図7】



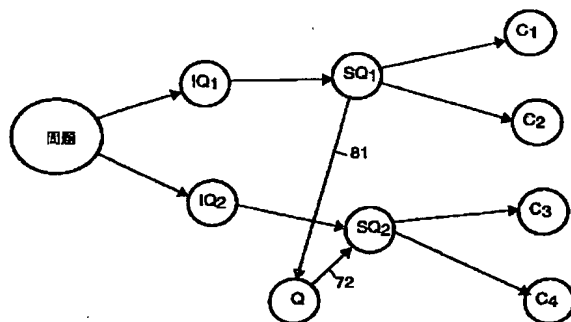
【図9】



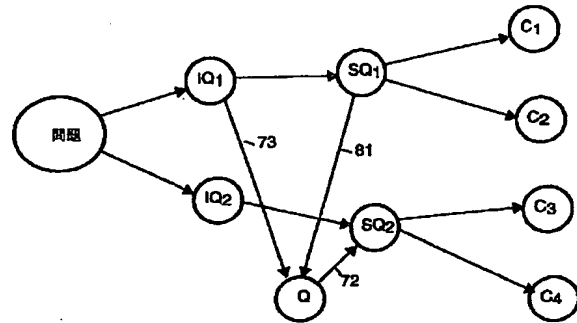
【図8】



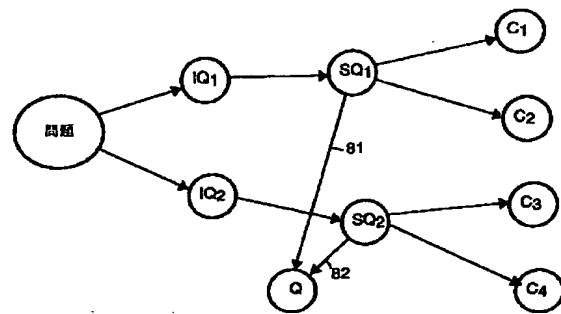
【図12】



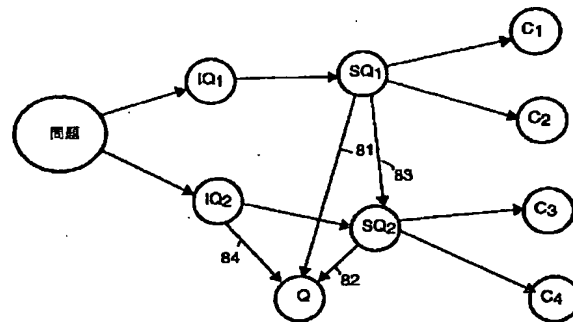
【図11】



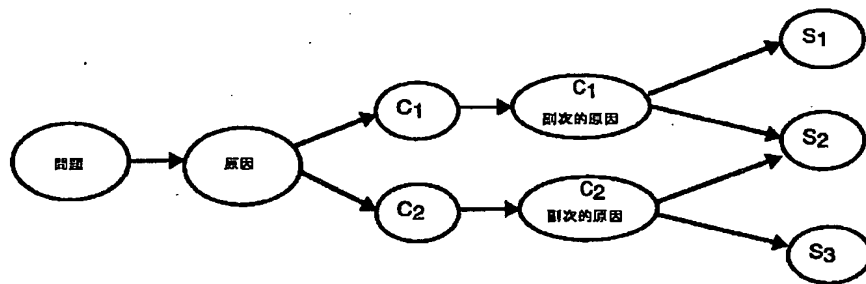
【図14】



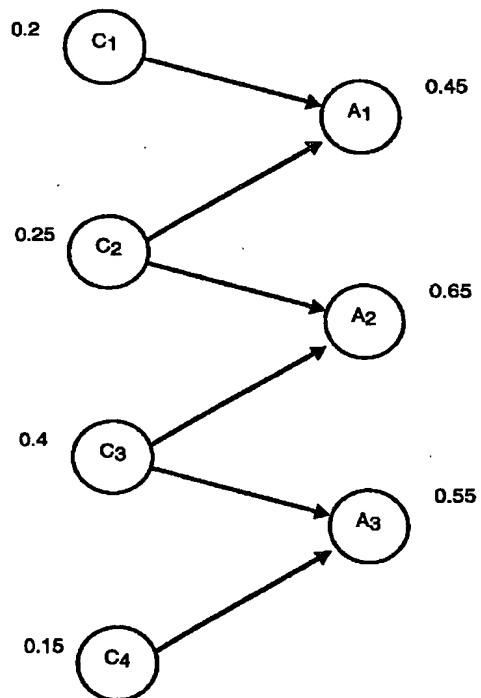
【図13】



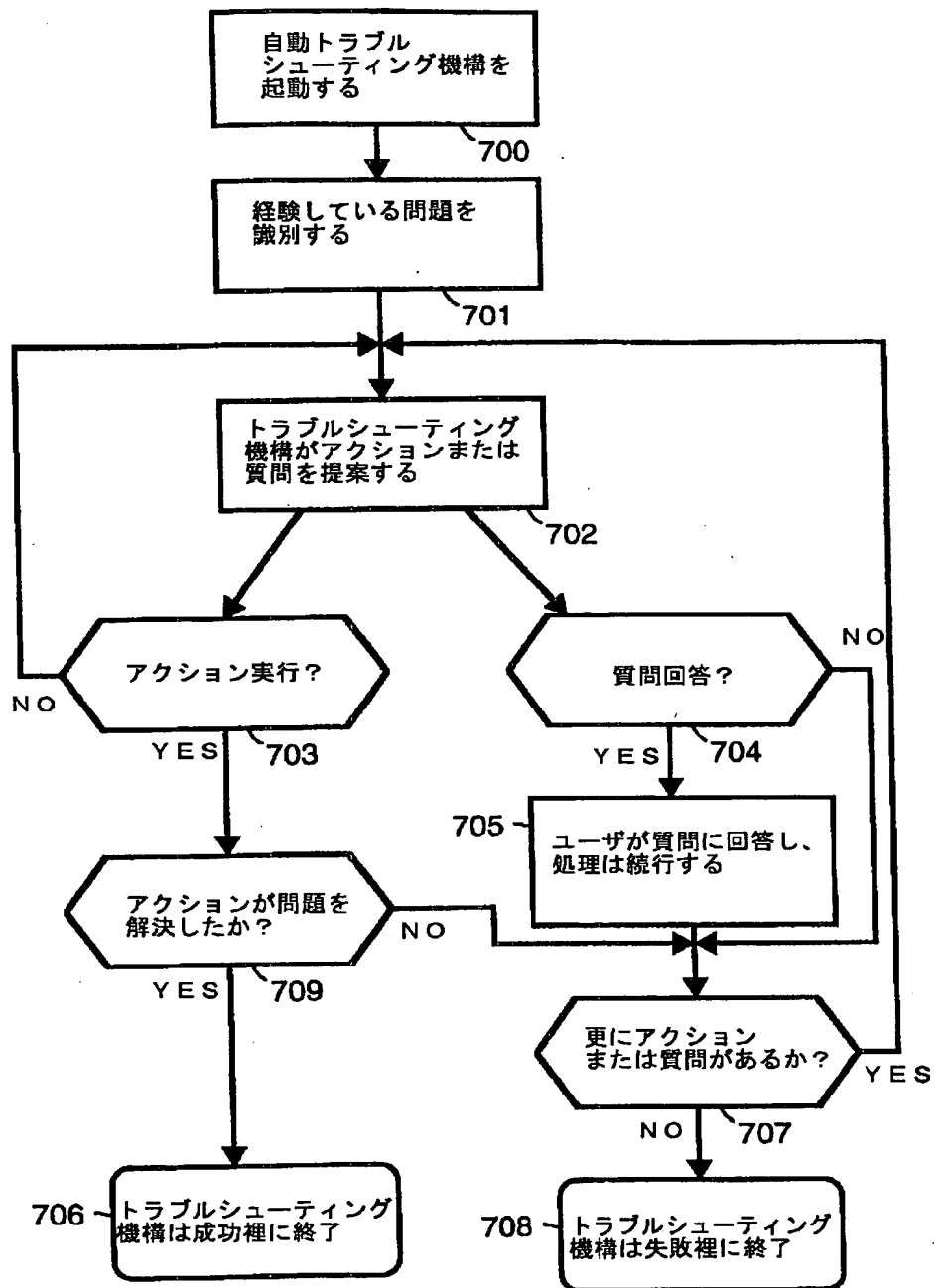
【図15】



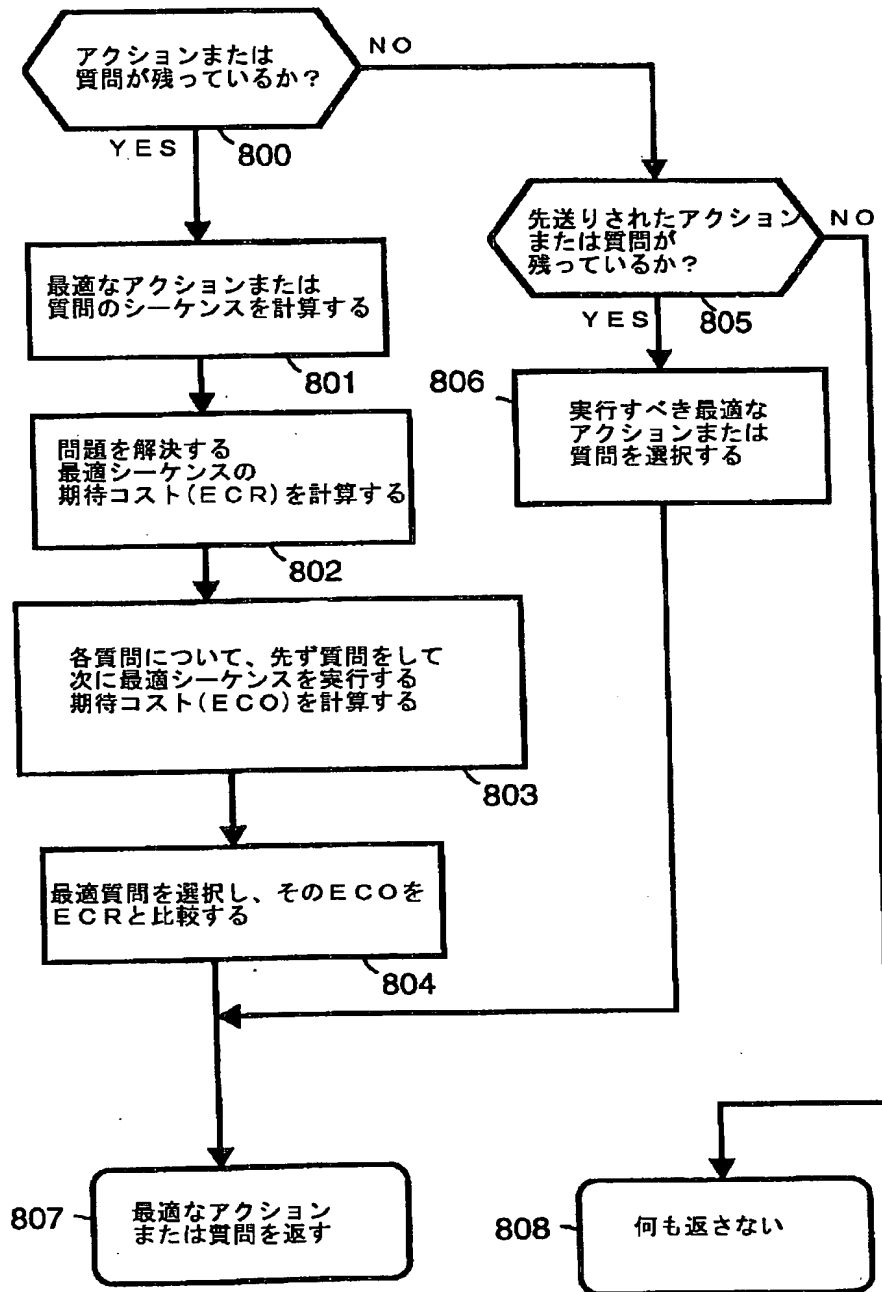
【図18】



【図16】



【図17】



フロントページの続き

(72)発明者 フィン・ヴィー・ジェンセン  
デンマーク、ディーケー—9700、プロエン  
デルスレフ、オルフスゲ—デ 26

(72)発明者 ユツフェ・カジャルルフ  
デンマーク、ディーケー—9220、アールボ  
ーグ、ホルドー・ラックスネス・ヴァイ

(72)発明者 ポール・エー・ペルティエ  
アメリカ合衆国83704アイダホ州ボイジー、  
ノース・コール・ロード 1800 アパート  
メント エル306

(72)発明者 ラッセ・ロストラップ・ジェンセン  
デンマーク、9000、アールボーグ、クリス  
チャンズゲーデ 1 ビー1 フロアー

(72)発明者 マリリン・エー・パーカー  
アメリカ合衆国83705アイダホ州ボイジー、  
サウス・イーグルサン・ロード 311

(72)発明者 ジャニス・エル・ボゴラッド  
デンマーク、3450、アレロド、エクレイ  
ーイ 20



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**